Universität Leipzig
Fakultät für Mathematik und Informatik
Institut für Informatik

Verbesserung einer Erkennungs- und Normalisierungsmaschine für natürlichsprachige Zeitausdrücke

Masterarbeit

Leipzig, Dezember 2012

vorgelegt von Stefan Thomas
Studiengang Informatik

Betreuender Hochschullehrer: Prof. Dr. Gerhard Heyer
Fakultät für Informatik und Mathematik, Institut für Informatik, Abteilung Automatische Sprachverarbeitung

Inhaltsverzeichnis

1 Einleitung

1.1 Motivation

Sowie Computer eine immer größer werdende Rolle im Leben einnehmen, so bekommen auch digital gespeicherte Daten und elektronische Kommunikationskanäle immer mehr Aufmerksamkeit und erfreuen sich einer stetig steigenden Verwendung. Dies gilt umso mehr im Hinblick auf die rasante Verbreitung mobiler Endgeräte wie Smart-Phones und Tablet-PCs. Zuletzt stieg beispielsweise der Absatz von Smart-Phones, um 47% [CGG+12]. Damit gehen auch die Probleme einher, die eine wachsende Digitalisierung und Nutzung von Mobilgeräten mit sich bringen: Klassische Bedienkonzepte sind nicht ohne weiteres auf diese Gerätetypen übertragbar. Die Entwicklung neuartiger Konzepte erfordert ein tiefgreifendes Verständnis über die Struktur und den Inhalt der Daten des Nutzers. Nur dadurch ist es möglich, E-Mails, SMS und andere Beiträge in elektronischen Kommunikationskanälen auszuzeichnen, um so den Nutzer bei der Erstellung von Kalendereinträgen und Erinnerungen zu unterstützen, oder diese Prozesse sogar komplett zu automatisieren. Auch anspruchsvollere Nutzungsmöglichkeiten sind denkbar. Sofern man die Gesamtheit der Daten eines Nutzers analysiert und sämtliche Zeitangaben versteht, kann man eine semantische Suche anbieten. Der Nutzer muss sich so nicht an einen konkreten Wortlaut erinnern, wenn er nach einer bestimmten Information sucht, sondern kann frei auf einem Zeitstrahl navigieren und sich alle, mit einem spezifischen Zeitpunkt oder mit einer Zeitspanne verknüpfte Daten anzeigen lassen.

Das Identifizieren von Bestandteilen eines Textes, die Zeitangaben bezeichnen, ist daher ein wichtiger Schritt, um neuartige Bedienkonzepte zu ermöglichen. Diesen Vorgang nennt man *Zeiterkennung*. Im Anschluss an die Erkennung einer Zeitangabe folgt im Regelfall ein Normalisierungsschritt, bei dem der Zeitausdruck semantisch untersucht wird. Das Ziel ist es, eine kanonische und eindeutige sowie vor allem korrekt maschinenverarbeitbare Repräsentation zu erreichen. Hierfür eignet sich beispielsweise der internationale ISO-Standard ISO 8601 [I06]. Dieser Vorgang wird *Zeitnormalisierung* (kurz: Normalisierung) genannt. Im weiteren Verlauf der Arbeit wird der Begriff *Zeiterkennung* synonym zu *Zeiterkennung und -normalisierung* verwendet. Ausnahmen ergeben sich entweder aus dem Kontext oder sind entsprechend gekennzeichnet.

1.2 Mehrsprachigkeit

Es existieren zahlreiche Strategien, um Zeiterkennung durchzuführen [AU10, B02, DM09, GS10]. Grob lassen sie sich in zwei Klassen einteilen: lernbasiert und regelbasiert. Lernbasierte Verfahren versuchen das sprachspezifische Wissen, welches für eine korrekte Erkennung und vor allem für die Normalisierung notwendig ist, durch Analyse von Beispielen zu extrahieren. Im Gegensatz dazu, wird dieses Wissen bei regelbasierten Verfahren vorab bereitgestellt. Die Regeln reagieren dann im Text auf vorgegebene Muster und überführen sie in das kanonische Repräsentationsformat.

Obwohl bereits viel Forschung auf diesem Gebiet betrieben wurde, werden fast ausschließlich einsprachige Systeme präsentiert [AU10, DM09]. Eine Ausnahme bildet [B02]. Es verfolgt überwiegend einen lernbasierten Ansatz, der prinzipiell für jede Sprache funktioniert. Als Eingabe wird hierzu ein mit Zeitangaben annotiertes Korpus benötigt, das heißt eine große Textsammlung, bei der sämtliche Zeitausdrücke ausgezeichnet sind. Vollautomatisch werden daraus Regeln abgeleitet, die später über reguläre Ausdrücke auf unbekannte Texte angewendet werden. Das Verfahren hat allerdings eine Reihe von Unzulänglichkeiten. So funktioniert es beispielsweise nur für Sprachen, bei denen Wörter durch Leerzeichen getrennt sind und die Wortstellung ziemlich genau dem Englischen oder dem Französischen entspricht. Ein weiteres Defizit ist die fehlende Berücksichtigung von Komposition[1]. Alle anderen Herangehensweisen verfolgen hauptsächlich einen regelbasierten Ansatz, der im Allgemeinen nur auf eine oder wenige Sprachen anwendbar ist.

TRIPS bzw. TRIOS [AU10] benutzt für die Zeiterkennung Conditional Random Fields und den TRIPS Parser [ABS08]. Dieser ist hochgradig sprachabhängig, da er zum Beispiel WordNet[2] nutzt. Für die regelbasierte Normalisierung werden reguläre Ausdrücke herangezogen. Die Maschine ist dadurch in der Lage, Datumsangaben, Zeitangaben, Zeitdauern und wiederkehrende Ereignisse zu erkennen und zu normalisieren – jedoch ausschließlich auf englischen Texten. Auch DANTE [DM06, DM08, DM09] basiert weitgehend auf einem umfangreichen Regelsystem, welches 252 manuell erstellte Regeln enthält. Diese überführen die vorgegebenen Muster zunächst in eine lokale Semantik. In einem weiteren Schritt werden die erkannten Zeitausdrücke bzgl. des Kontextes

[1] Komposition bezeichnet die Bildung eines neuen Wortes durch Konkatenation vorhandener Wörter.

[2] WordNet ist ein großes maschinenlesbares Lexikon für Englisch: http://wordnet.princeton.edu/

interpretiert. Dabei entsteht schließlich eine globale Semantik, das heißt die kanonische Repräsentationsform. HeidelTime [GS10] führt erstmals eine strikte Trennung zwischen sprachspezifischen Wissen und dem Algorithmus zur Zeiterkennung ein. Der sprachunabhängige Teil gibt nur die Architektur vor – selbst die Regeln können (und müssen) bzgl. einer bestimmten Sprache spezifiziert werden. HeidelTime funktioniert dadurch gegenwärtig zwar sowohl für Englisch und Deutsch als auch für Niederländisch und lässt sich ohne Programmierkenntnisse auf weitere Sprachen erweitern, jedoch ist hierfür die Bereitstellung von sehr umfangreichen sprachspezifischen Wissen nötig.

Aufgrund der Vielzahl an verschiedenen Sprachen, sind alle bisherigen Ansätze nicht auf eine große Anzahl von Sprachen skalierbar oder zumindest wäre dies nur mit extrem großem manuellen Aufwand möglich. Eine umfangreiche Sprachunterstützung ist indessen für eine Markt- und Nutzerakzeptanz erforderlich. Die Entwicklung eines Verfahrens, welches weitgehend sprachunabhängig arbeitet, ist daher notwendig. Im Rahmen seiner Tätigkeit bei ExB Research & Development GmbH hat Herr Christian Hänig Anfang 2012 eine Maschine implementiert, die nach diesem Prinzip arbeitet. Die Erkennungsrate und die Normalisierungsfähigkeit der bestehenden Zeiterkennungsmaschine sind allerdings eingeschränkt. Abb. 1 ist zu entnehmen, dass insbesondere die Normalisierung nur in jedem zweiten Fall korrekt ist. Das Evaluations-Verfahren zur Ermittlung der Werte sowie die Evaluationsmaße selbst werden in Kapitel 4 betrachtet.

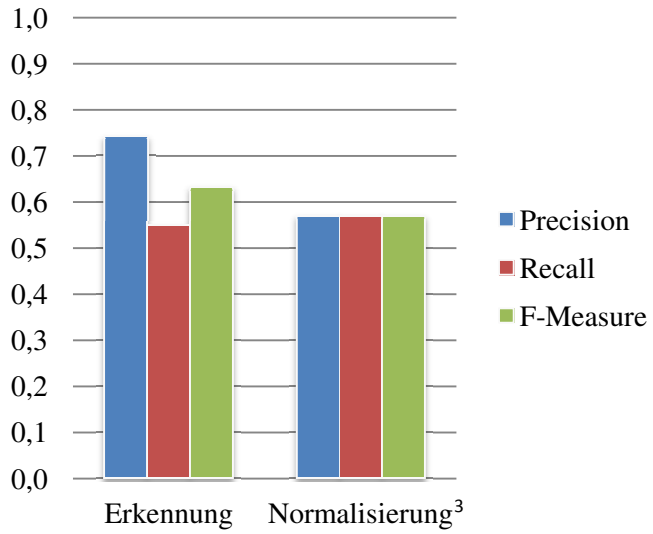

Abb. 1: Übersicht über Fähigkeit der ursprünglichen Implementierung

[3] Aufgrund der Evaluationsmethodik (siehe Abschnitt 4.2) und der Architektur der Zeiterkennungsmaschine gilt für die Normalisierung, dass Precision und Recall stets den gleichen Wert aufweisen.

Dass sowohl die Erkennung als auch der Normalisierungsschritt – speziell unter Berücksichtigung einer Mehrsprachigkeit – nicht trivial sind, sticht sofort ins Auge, wenn man bedenkt wie differenziert Zeitausdrücke aussehen können. Abb. 2 zeigt exemplarisch in drei verschiedenen Sprachen jeweils die gleichen vier temporalen Ausdrücke, die zudem alle dieselbe Semantik aufweisen.

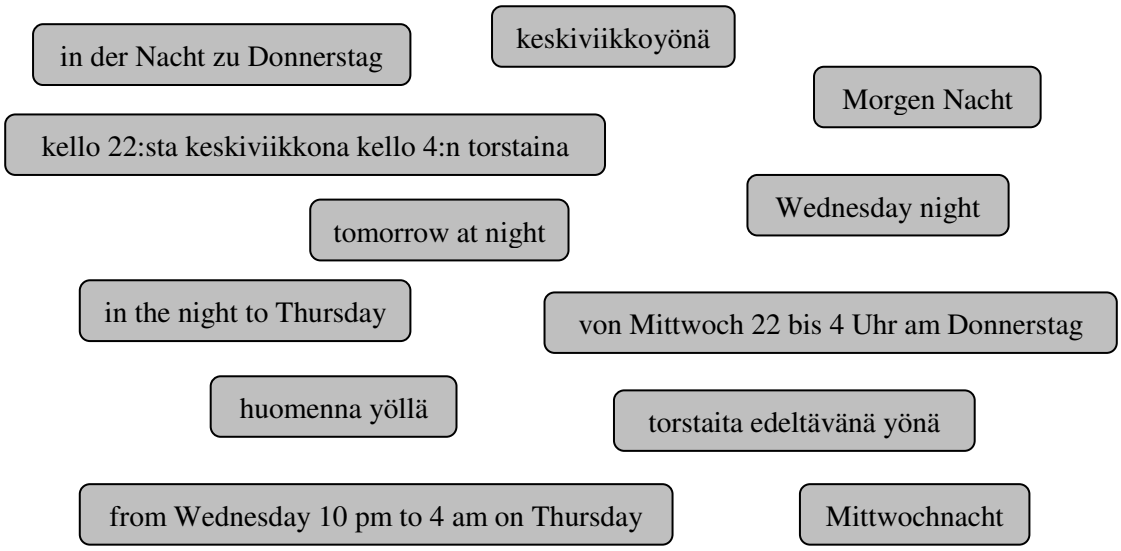

Abb. 2: Beispiele für natürlichsprachige Zeitausdrücke

1.3 Einordnung Zeiterkennung und -normalisierung

Das Erkennen von Zeitangaben in natürlichsprachigen Texten ist ein Spezialfall der sogenannten Named Entity Recognition, welche wiederum eine Teilaufgabe der Informationsextraktion darstellt. Unter Named Entity Recognition (kurz: NER) versteht man die Identifikation von Elementen innerhalb von natürlichsprachigen Texten, die sich in bestimmte Kategorien einordnen lassen. Es gibt eine Vielzahl solcher Kategorien, wie zum Beispiel Personen, Örtlichkeiten, Firmen und andere Organisationen, oder auch Proteine und Krankheiten. Diese Elemente werden *benannte Entitäten* genannt.

Zur Informationsextraktion gehört neben der Erkennung und Klassifizierung der benannten Entitäten ebenso die Extraktion von Beziehungen zwischen Entitäten und von Ereignissen [G03]. Beispielsweise besteht in dem Satz „Jakob ist in der Schule.“ eine Beziehung zwischen den Entitäten *Jakob* und *Schule*, welche besagt, dass sich die *Person* mit dem Namen *Jakob* an einem *Ort Schule* aufhält. Die Ermittlung solcher Beziehungen zwischen Zeitangaben und anderen Entitäten, sowie die beiden aufeinander aufbauenden Schritte Zeiterkennung und Zeitnormalisierung selbst, sind Voraussetzung für

eine Reihe weiterer Aufgaben des Information Retrieval und des Natural Language Processing oder bieten zumindest eine Unterstützung [ABG07, CCG+09, DM06, FMS+01, KLP+05]. Als Beispiele seien hier Question Answering, Text Summarization und generell das Verstehen eines Textes genannt. Aufgrund des Prinzips *Garbage In – Garbage Out* [BHL10], ist eine präzise Bewältigung der Zeiterkennung und der anschließenden Normalisierung essentiell. Das Prinzip besagt, dass ein Computer die Eingabe unsinniger Daten nicht hinterfragt und aus diesen wiederum unsinnige Ausgaben erzeugt.

Üblicherweise werden die Texte im Rahmen von NER um Meta-Daten ergänzt, damit diese später durch andere Verfahren oder Anwendungen verwendet werden können. Die sogenannten Annotationen[4] beziehen sich auf die identifizierten Textbestandteile und enthalten bei Zeitangaben als Attribute die kanonische Form des temporalen Ausdrucks.

In der Vergangenheit fand eine Vielzahl von Wettbewerben statt, bei denen unter anderem die Aufgabe gestellt wurde, Zeitausdrücke zu erkennen und zu normalisieren. Ende der neunziger Jahre hat die DARPA[5] die Message Understanding Conferences (MUC) initiiert, bei denen ausschließlich Zeiterkennung betrachtet wurde. Zeitnormalisierung ist erst einige Jahre später hinzugekommen und zwar im Rahmen der Workshops innerhalb des Automatic Content Extraction (ACE) Programms[6]. Im Laufe der Zeit ist zudem auch die Aufgabe deutlich schwieriger geworden, da eine immer größere Anzahl von Typen temporaler Ausdrücke mit einbezogen wurde. Zuletzt haben acht Teams mit insgesamt 15 Maschinen an einem weiteren Wettbewerb – dem SemEval-2010[7] – teilgenommen. Darunter befinden sich neben den Entwicklern von TRIPS und TRIOS auch die Urheber von HeidelTime. Die veröffentlichten Evaluationsergebnisse für Englisch [CPS+10] mit durchschnittlichen F-Measure-Werten von 0,78 für die Zeiterkennung und 0,61 für die Zeitnormalisierung unterstreichen insbesondere die Schwierigkeit der Normalisierung. In diesem Zusammenhang ist ebenfalls das in [LPS+06] ermittelte Inter-Annotator-Agreement[8] von 0,9 zu erwähnen.

[4] Annotationen können sowohl von automatischen Algorithmen als auch manuell von Menschen, den sogenannten Annotatoren, angelegt werden. Letzteres ist insbesondere für die Erstellung eines Gold-Standards relevant, welcher für die Evaluation der Algorithmen notwendig ist.

[5] Defense Advanced Research Projects Agency

[6] http://www.nist.gov/speech/tests/ace

[7] TempEval-2 Task 13: http://semeval2.fbk.eu/semeval2.php?location=tasks&area=Time%20Expressions

[8] Inter-Annotator-Agreement bezeichnet die Übereinkunft mehrerer Annotatoren bzgl. ihrer erstellten Annotationen und ist ein Maß dafür wie schwer eine Annotationsaufgabe ist bzw. welche Erwartung man an einen automatischen Algorithmus maximal stellen kann, der die selbe Aufgabe lösen soll.

An dieser Stelle sei außerdem der Einfluss des verwendeten Kalendersystems konstatiert. Es existieren viele unterschiedliche Kalendersysteme, jedoch ist weltweit vor allem der Gregorianische Kalender verbreitet. Dieser wurde 1582 eingeführt, um im Vergleich zum Julianischen Kalender eine bessere Approximation an das Tropische Jahr zu erreichen. Beides sind Vertreter von Solarkalendern, bei denen die Lage der Erde zur Sonne die astronomische Grundlage bildet. Andere Kalendersysteme basieren auf der Bewegung des Mondes und werden daher Lunarkalender genannt, oder sind – im Falle von Lunisolarkalendern – eine Mischform, die auf den beiden erstgenannten Prinzipien aufbauen. Im weiteren Verlauf und insbesondere bei der Definition der Zeitskala in Abschnitt 2.1 wird lediglich der Gregorianische Kalender berücksichtigt.

1.4 Ziel dieser Arbeit

Das Ziel ist es, Strategien zu entwickeln, die sowohl zu einer Verbesserung der Zeiterkennung als auch zu einer Steigerung der Genauigkeit der Zeitnormalisierung führen. Dabei sollen insbesondere Ressourcenrestriktionen bzgl. Speicher und Prozessorleistung sowie eine weitgehende Sprachunabhängigkeit Berücksichtigung finden, aber dennoch eine unverzügliche Verarbeitung von kurzen und mittellangen Texten, wie zum Beispiel einer SMS oder einer E-Mail, ermöglicht werden.

1.5 Gliederung

Die Arbeit ist wie folgt gegliedert: In Kapitel 2 werden grundlegende Definitionen eingeführt und die Ausgangslage detailliert analysiert. Dabei werden zum einen typische Beispiele für temporale Ausdrücke vorgestellt und zum anderen die Architektur der Maschine betrachtet. Ausgehend von den gewonnenen Erkenntnissen werden in Kapitel 3 Möglichkeiten zur Verbesserung vorgestellt. Die entwickelten Verfahren werden in Kapitel 4 anhand manuell annotierter Texte für verschiedene Sprachen evaluiert. Im abschließenden Kapitel wird eine Zusammenfassung gegeben und es werden Weiterentwicklungsmöglichkeiten aufgezeigt.

2 Analyse

In diesem Kapitel werden zunächst grundlegende Begrifflichkeiten eingeführt. Ausgehend von einer detaillierten Analyse der Ausgangslage und den Anforderungen an die Zeiterkennungsmaschine werden Ansatzpunkte für notwendige und vielversprechende Verbesserungen ermittelt. Weiterhin werden sowohl innerhalb eines temporalen Ausdrucks liegende als auch ausdrucksübergreifende Probleme und Zusammenhänge betrachtet, welche bei der Entwicklung von Verbesserungsansätzen berücksichtigt werden müssen. Als Ausgangsdaten dienen hierbei jeweils 25 deutsche und englische Texte mit insgesamt 204 temporalen Ausdrücken.

2.1 Definitionen

In der Linguistik wird unter anderem versucht eine Theorie über die Bedeutung sprachlicher Ausdrücke zu entwerfen, d.h. die Semantik von natürlicher Sprache zu erfassen und zu formalisieren. In diesem Zusammenhang werden formale Logiksprachen herangezogen, in welche die natürlichsprachigen Ausdrücke übersetzt werden. Beispiele für Logiksprachen sind die Aussagenlogik, die Prädikatenlogik und die in [M74] formulierte intensionale Logik. Letztgenannte beinhaltet auch die Modellierung von Zeit in natürlicher Sprache, welcher in einem Teilgebiet der Linguistik, der sogenannten Temporalsemantik, nachgegangen wird. In diesem Zusammenhang existieren auch spezifische temporale Modelle [L11, M74, R47]. In Anlehnung daran werden hier notwendige Definitionen und Konzepte eingeführt, die sich zum Teil aber auch deutlich von diesen abgrenzen.

Def. temporale Einheit: Eine temporale Einheit ist eine Maßeinheit um Zeit anzugeben. Im Folgenden sei dies beschränkt auf: Millisekunde (ms), Sekunde (s), Minute (m), Stunde (h), Tag (d), Monat (M) und Jahr (y). Die Menge aller temporalen Einheiten sei mit U bezeichnet, das heißt $U = \{'ms', 's', 'm', 'h', 'd', 'M', 'y'\}$. Auf U besteht eine lineare Ordnung mit $'ms' < 's' < 'm' < 'h' < 'd' < 'M' < 'y'$. Sei $u \in U$. $u - 1$ bezeichnet dann die größte temporale Einheit, welche kleiner als u ist und $u + 1$ die kleinste temporale Einheit, welche größer als u ist. $'ms' - 1$ und $'y' + 1$ sind dabei undefiniert.

Def. Zeitpunkt: Ein Zeitpunkt t ist ein Moment innerhalb eines zeitlichen Bezugssystems. Er besitzt keine Länge und lässt sich auf einer Zeitskala darstellen. Die Zeitskala besitzt einen Nullpunkt und kann auch als Zeitstrahl oder Zeitachse bezeichnet werden. Es besteht eine lineare Ordnung. Ein Zeitpunkt sei im Folgenden durch die Angabe von Millisekunde, Sekunde, Minute, Stunde, Tag, Monat und Jahr definiert[9], das heißt $t = (ms, s, m, h, d, M, y)$ mit $ms \in [0{,}999], s, m \in [0, 59]$, $h \in [0, 23], d \in [1, 31], M \in [1, 12], y \in \mathbb{Z}$. Die Menge aller Zeitpunkte sei mit Z bezeichnet. Für den Zugriff auf einzelne Werte eines Zeitpunkts sei verkürzend die Notation $t['ms']$, $t['s']$ usw. erlaubt. Sei $x \in [0, 999]$. $t_2 = t['ms' = x]$ ist dann verkürzend für $t_2 = (x, t['s'], t['m'], t['h'], t['d'], t['M'], t['y'])$.

Def. Minimum- und Maximum-Funktion: Sei $t = (ms, s, m, h, d, M, y) \in Z$ ein Zeitpunkt. Die Funktion $min: Z \times U \longrightarrow \mathbb{N}_0$ ist auf allen Zeitpunkten und temporalen Einheiten definiert und gibt das erlaubte Minimum der jeweiligen temporalen Einheit bzgl. t wieder. Analog ist $max: Z \times U \longrightarrow \mathbb{N}_0$ definiert, jedoch bildet diese Funktion auf das erlaubte Maximum ab.

Def. Zeitspanne: Seien t_1 und t_2 zwei Zeitpunkte mit $t_1 < t_2$. Eine Zeitspanne ist dann das Intervall $[t_1, t_2]$. Eine Zeitspanne hat somit im Gegensatz zum Zeitpunkt eine Ausdehnung. t_1 wird als Startzeitpunkt und t_2 als Endzeitpunkt bezeichnet. Die Menge aller Zeitspannen sei mit S bezeichnet. Für den Zugriff auf den Start- bzw. Endzeitpunkt sei die Notation $[t_1, t_2].1$ respektive $[t_1, t_2].2$ erlaubt. Zeitintervall (kurz: Intervall) ist synonym zu Zeitspanne. Abzugrenzen ist hier allerdings der Begriff Zeitdauer, siehe nächste Definition.

Def. Zeitdauer: Eine Zeitdauer d hat ähnlich zur Zeitspanne eine Ausdehnung, aber diese weist keinen konkreten Bezug zu der eingangs eingeführten Zeitskala auf. Es gilt $d = (ms, s, m, h, d, M, y)$ mit $ms, s, m, h, d, M, y \in \mathbb{N}$. Die Menge aller Zeitdauern sei mit D bezeichnet.

Def. temporaler Ausdruck: Ein temporaler Ausdruck oder Zeitausdruck (im Folgenden auch kurz: Ausdruck) besteht im Allgemeinen aus einem oder mehreren natürlichsprachigen Wörtern und/oder einer Kombination von Ziffern und Zeichen, deren

[9] In Beispielen wird aus Gründen der Übersicht meist auf die Angabe der Millisekunden verzichtet. In diesem Fall ergibt sich aus dem Kontext, ob 0 oder 999 Millisekunden gemeint sind.

Semantik einen zeitlichen Bezug aufweist. Dieser Bezug kann in Form eines oder mehrerer Zeitpunkte, Zeitspannen oder Zeitdauern gegeben sein.

Def. Bezugszeitpunkt: Sei a ein temporaler Ausdruck. Der Bezugszeitpunkt $b \in Z$ gibt den Zeitpunkt an, welcher als Basis für die Interpretation von a verwendet werden muss.

Def. Referenzzeitpunkt: Sei T die Menge aller Texte beliebiger Länge und sei $\tau \in T$. Der Referenzzeitpunkt $r \in Z$ gibt den Zeitpunkt an, zu dem τ verfasst bzw. veröffentlicht oder verschickt wurde. Er trägt eine besondere Bedeutung bei der Interpretation von Zeitausdrücken innerhalb von τ, da er für diese im Allgemeinen als Bezugszeitpunkt dient.

Def. Teiltext-Funktion: Sei $\tau \in T$ ein Text und bezeichne len_τ die Länge von τ hinsichtlich Anzahl der Zeichen. Weiterhin seien $p_1, p_2 \in \mathbb{N}$. Die Teiltext-Funktion $s: T \times \mathbb{N} \times \mathbb{N} \longrightarrow_p T$ ist eine partielle Funktion mit $s(\tau, p_1, p_2)$ genau dann definiert, wenn τ ein nichtleerer Text ist und $1 \leq p_1 \leq p_2 \leq len_T$. $s(\tau, p_1, p_2)$ bezeichnet genau den Teiltext von τ, welcher an Zeichenposition p_1 beginnt und direkt nach p_2 endet. Die erste Zeichenposition eines Textes ist 1.

Das Zeiterkennungs-Problem lässt sich nun folgendermaßen formalisieren:

> *Gegeben sei ein Text $\tau \in T$. Finde die größte Menge M, so dass für alle $(p_1, p_2) \in M$ mit $p_1, p_2 \in \mathbb{N}$ gilt: $s(\tau, p_1, p_2)$ ist definiert und ein temporaler Ausdruck und $\nexists o_1, o_2 \in \mathbb{N}$ mit $s(\tau, p_1 - o_1, p_2 + o_2)$ ist definiert und ein temporaler Ausdruck. M ist dann Antwort auf das Zeiterkennungs-Problem.*

Def. schwache Interpretationsfunktion: Sei a ein temporaler Ausdruck. $\bar{\iota}: T \longrightarrow \mathcal{P}(U)$ ist eine Funktion mit $\bar{\iota}(a)$ bezeichnet die Menge der in a spezifizierten temporalen Einheiten.

Def. (starke) Interpretations-Funktion: Seien $\tau \in T$, $r \in Z$ *der Referenzzeitpunkt von* τ sowie $p_1, p_2 \in \mathbb{N}$. $i: T \times \mathbb{N} \times \mathbb{N} \times Z \longrightarrow_p \mathcal{P}(Z \times \mathbb{N}) \cup \mathcal{P}(S \times \mathbb{N}) \cup \mathcal{P}(D \times \mathbb{N})$ ist eine partielle Funktion mit $i(\tau, p_1, p_2, r)$ genau dann definiert, wenn $s(\tau, p_1, p_2)$ definiert und ein temporaler Ausdruck ist. $i(\tau, p_1, p_2, r) = \{(v_1, g_1), (v_2, g_2), \dots\}$ bezeichnet dann die Semantik des Ausdrucks bzw. seine kanonische Repräsentationsform. Gleiche Werte für g_1, g_2 usw. bedeuten, dass die entsprechenden Zeitpunkte,

Zeitspannen oder Zeitdauern Alternativen zueinander darstellen. Alternativen sind immer dann nötig, wenn sonst keine eindeutige Interpretation möglich ist. Unterschiedliche Werte stehen dagegen für multiple Zeitpunkte, Zeitspannen oder Zeitdauern. Die konkreten Werte für g_1, g_2 usw. haben sonst keine weitere Bedeutung.

Schließlich lässt sich auch das Zeitnormalisierungs-Problem formalisieren:

> *Gegeben seien ein Text $\tau \in T$, die Antwort auf das Zeiterkennungs-Problem M und $r \in Z$ der Referenzzeitpunkt von τ. Finde die größte Menge N, so dass für alle $\big((p_1, p_2), n\big) \in N$ mit $(p_1, p_2) \in M$ gilt: $i(\tau, p_1, p_2, r)$ ist definiert und $n = i(\tau, p_1, p_2, r)$. N ist dann Antwort auf das Zeitnormalisierungs-Problem.*

Im Folgenden seien a ein temporaler Ausdruck und τ ein Text mit $a = s(\tau, p_1, p_2)$ für geeignete $p_1, p_2 \in \mathbb{N}$. Weiterhin seien $n = \{(v_1, g_1), (v_2, g_2), \dots\}$ die kanonische Repräsentationsform von a, und $\bar{n}$ das Ergebnis der schwachen Interpretationsfunktion. a weist genau dann einen *konkreten Zeitachsenbezug* auf (kurz: a *ist konkret*), wenn $n \in \mathcal{P}(Z \times \mathbb{N}) \cup \mathcal{P}(S \times \mathbb{N})$. Ansonsten weist a *keinen konkreten Zeitachsenbezug* auf bzw. ist *nicht konkret*. Ein konkreter Ausdruck a ist genau dann *relativ*, wenn die Elemente von n, unabhängig vom Bezugszeitpunkt b, entweder stets in der Zukunft oder stets in der Vergangenheit von b liegen. Ansonsten ist a *absolut*. a ist genau dann *vollständig spezifiziert*, wenn a absolut ist sowie $\forall u \in U \setminus \{'y'\}: u \in \bar{n} \Rightarrow u + 1 \in \bar{n}$. Falls die zweite Bedingung nicht zutrifft, so ist a *unterspezifiziert*.

Für die Interpretation von vollständig spezifizierten Ausdrücken ist somit in der Regel kein zusätzliches Wissen notwendig. Im Gegensatz dazu wird für eine korrekte Normalisierung von unterspezifizierten und relativen Ausdrücken in der Regel Kontextwissen benötigt. In diesem Zusammenhang ist vor allem der Bezugszeitpunkt hilfreich. Dieser kann – muss aber nicht – dem Referenzzeitpunkt r entsprechen.

a ist genau dann *exakt*, wenn für alle $x, y \in [1, |n|]$ mit $x \neq y$ gilt: $g_x \neq g_y$, das heißt, dass kein Element der kanonischen Repräsentationsform eine Alternative zu einem anderen Element darstellt. Ansonsten ist a *ungenau*. a beschreibt genau dann ein *wiederkehrendes Ereignis*, wenn $|n| > 1$ und es existieren $x, y \in [1, |n|]$, so dass $g_x \neq g_y$.

Zur Verdeutlichung dieser Eigenschaften siehe Tab. 1. Diese zeigt exemplarisch zehn Sätze mit jeweils einem temporalen Ausdruck. Demnach weist der Ausdruck „Am

21.12.2012“ einen konkreten Zeitachsenbezug auf, ist absolut, vollständig spezifiziert und exakt. Im Gegensatz dazu ist „Am 8. Mai“ zwar auch konkret, absolut und exakt, jedoch unterspezifiziert, da die Jahresangabe fehlt. Der dritte Beispielsatz beinhaltet einen relativen Zeitausdruck, welcher aber im Unterschied zum vierten Beispiel zumindest exakt ist. Der Ausdruck „fünf Jahre“ im darauf folgenden Beispielsatz weist erstmals keinen Zeitachsenbezug auf. Es sind noch viele weitere Kombinationen von Eigenschaften möglich und denkbar. Einige davon sind ebenfalls in Tab. 1 angegeben.

Beispiel	Eigenschaft
„**Am 21.12.2012** geht die Welt unter.“	konkret, absolut, vollständig spezifiziert, exakt
„**Am 8. Mai** habe ich einen Termin.“	konkret, absolut, unterspezifiziert, exakt
„Wir treffen uns **in 5 Stunden**.“	konkret, relativ, exakt
„Ich laufe **an einem der folgenden Tage**.“	konkret, relativ, ungenau
„Ich bin **fünf Jahre** alt.“	nicht konkret, exakt
„Das war **irgendwann im Juni**.“	konkret, absolut, unterspezifiziert, ungenau
„**Im Dezember** war ich im Urlaub.“	konkret, absolut, unterspezifiziert, exakt
„Das dauert **mindestens drei Tage**.“	nicht konkret, ungenau
„Rühren Sie **ca. 5 Minuten später** gut um.“	konkret, relativ, ungenau
„Es findet **jeden Mittwoch im Juni** statt.“	konkret, absolut, unterspezifiziert, exakt, beschreibt wiederkehrendes Ereignis

Tab. 1: Temporale Ausdrücke zur Verdeutlichung von Eigenschaften

Unterspezifizierte, relative und ungenaue Ausdrücke wirken sich negativ auf die Qualität der Normalisierung aus. Es ist somit anzustreben, insbesondere Ausdrücke mit diesen Eigenschaften als solche zu erkennen und entsprechend zu behandeln. Dies bedeutet beispielsweise im Fall von unterspezifizierten Zeitausdrücken, die fehlenden Informationen aus dem Kontext zu ermitteln, oder bei relativen Zeitangaben den Bezugszeitpunkt festzustellen. Diese Prozesse unterstützen vor allem den Normalisierungs-Vorgang, können aber auch positiven Einfluss auf den Erkennungs-Vorgang haben und somit dabei helfen, die Qualität der Erkennung und der Normalisierung im Hinblick auf Precision und Recall zu verbessern.

2.2 Temporale Ausdrücke

Um vielversprechende Ansatzpunkte für eine Verbesserung zu finden, ist zunächst eine systematische Analyse temporaler Ausdrücke notwendig. In diese Analyse sind mehrere deutsche und englische Texte mit insgesamt etwa 200 temporalen Ausdrücken eingegangen.

2.2.1 Eigenschaften

In Tab. 2 sind die in Abschnitt 2.1 eingeführten Eigenschaften temporaler Ausdrücke und deren Häufigkeit aufgeführt. Das überproportionale Auftreten von Ausdrücken, die Zeitspannen beschreiben, ist auffällig. Ebenfalls erwähnenswert ist, dass sich jeder vierte Ausdruck auf einen Wochentag bezieht. Diese sind besonders schwierig zu interpretieren [B02, DM08] und werden daher in Abschnitt 2.2.2 eingehend untersucht. Abschnitt 2.2.3 widmet sich anschließend Datums- und Uhrzeitangaben, denn auch diese weisen eine hohe Auftretenshäufigkeit auf.

Eigenschaft	Häufigkeit	Eigenschaft	Häufigkeit
konkreter Zeitachsenbezug	180	kein Zeitachsenbezug	24
absolut	97	relativ	83
vollständig spezifiziert	28	unterspezifiziert	69
beschreibt Zeitpunkt	22	beschreibt Zeitspanne	158
exakt	168	ungenau	36
beschreibt Häufigkeit / wiederkehrendes Ereignis	4		
beinhaltet Bezug zu Wochentag	51		
beinhaltet Bezug zu Monatsname	11		
beinhaltet Bezug zu Tageszeit	10		
Datumsangabe	24		
Uhrzeitangabe	10		

Tab. 2: Häufigkeit von Eigenschaften temporaler Ausdrücke

2.2.2 Wochentage und Monatsnamen

Monatsnamen, aber insbesondere Namen von Wochentagen, treten relativ häufig in Texten auf. Beiden Ausdruckstypen liegt eine inhärente Problematik zugrunde: Sie sind nahezu immer unterspezifiziert. Das bedeutet, dass eine korrekte Interpretation dieser Ausdrücke nur mit Hilfe von Kontextwissen sichergestellt werden kann. Für Wochentage und Monatsnamen kommt dabei vor allem die Berücksichtigung der Zeitform des mit ihm im Zusammenhang stehenden Verbs in Betracht. In [DM08] und [ARR07] konnte gezeigt werden, dass dies – zumindest für Englisch – funktioniert und sehr präzise Ergebnisse liefert. Problematisch ist allerdings das umfassende sprachspezifische Wissen, welches dabei notwendig ist. Alternativ bieten sich einfache, aber ebenfalls häufig zum Ziel führende Heuristiken an. Ein Beispiel ist die Verwendung eines speziellen 7-Tage-Fensters für Namen von Wochentagen [B02]. Zwar ist dieser Ansatz der sprachspezifischen Analyse der Zeitform eines Verbs unterlegen, jedoch laut [DM08] nur sehr knapp. Entscheidend ist dabei die Wahl des Fensters, da sich hier verschiedene Möglichkeiten ergeben: Die letzten oder die nächsten sieben Tage oder ein Zeitraum, welcher sich über die letzten drei und die nächsten drei Tage erstreckt sowie die Entscheidung, ob der aktuelle Tag[10] zum Fenster dazu gehört oder nicht. Auf analoge Weise kann die Normalisierung von Monatsnamen behandelt werden.

Es ist indessen nicht klar, inwiefern die Art der Texte Einfluss auf den Erfolg der Heuristiken hat. Es ist anzunehmen, dass ein deutlicher Unterschied zwischen Zeitungstext, E-Mail-Korrespondenz und Kommunikation innerhalb von sozialen Netzwerken und Micro-Blogging Diensten wie Twitter herrscht. Auf eine weiterführende Betrachtung wird dennoch hier verzichtet, da es den Rahmen dieser Arbeit sprengen würde.

2.2.3 Datums- und Uhrzeitangaben

Speziell Datumsangaben kommen in einer großen Variationsvielfalt vor. Vollständig spezifizierte Datumsangaben sind meist problemlos zu erkennen. Sobald jedoch eine verkürzte Schreibweise verwendet wird, so dass es sich letztendlich um einen unterspezifizierten Zeitausdruck handelt, ist es deutlich schwerer eine echte Datumsangabe von anderen Dingen zu differenzieren. Die Berücksichtigung von sprachspezifischen Eigenheiten, wie zum Beispiel dem Dezimaltrennzeichen, kann hierbei Unterstützung bieten.

[10] *Aktueller Tag* meint hier bzgl. des Referenzzeitpunkts.

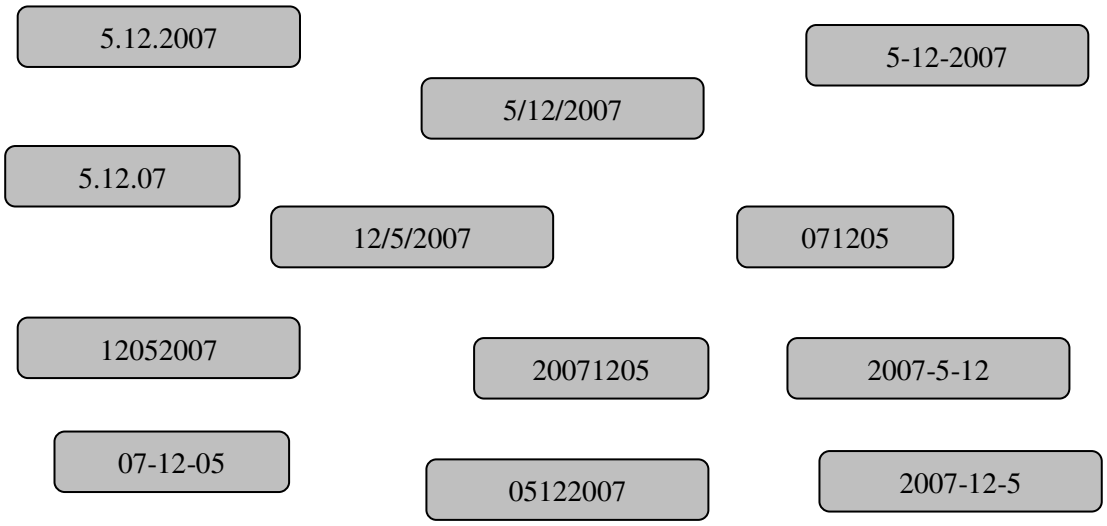

Abb. 3: Auswahl verschiedener Datumsformate und Schreibweisen

In Abb. 3 ist eine Auswahl verschiedener Datumsformate und Schreibweisen anhand des Datums *5. Dezember 2007* dargestellt. Es ist offensichtlich, dass insbesondere die Angabe des Tages und des Monates äußerst schwierig zu differenzieren sind, wenn keine Information darüber existiert, welche Reihenfolge bei der Angabe des Datums verwendet wird. Die Nutzung des üblichen Datumsformats des jeweiligen Kulturraums bzw. des Datumsformats, welches innerhalb der jeweiligen Sprache am meisten verbreitet ist, sollte hier angestrebt werden. Im Idealfall ist sogar das textspezifische Datumsformat zu ermitteln und anschließend zu verwenden.

Problematisch sind Ausdrücke wie „5.12“, da unklar ist, ob es sich dabei um eine Dezimalzahl oder um eine Datumsangabe handelt. Dies ist insbesondere in den Sprachen von Relevanz, bei denen der Punkt das offizielle Dezimaltrennzeichen ist. Das sprachspezifische Dezimaltrennzeichen sollte also bei der Erkennung von Datumsangaben eine entscheidende Rolle einnehmen.

Uhrzeitangaben sind im Vergleich zu Datumsangaben nicht so kompliziert. Zwar gibt es hier ebenso einige Variationsmöglichkeiten, jedoch deutlich weniger als bei Datumsangaben. Als Beispiele seien hier die Uhrzeitformate *h:mm*, *h-mm*, *h.mm*, *hhmm* und *h:mm:ss* genannt. Es wird schnell klar, dass auch hier das Dezimaltrennzeichen herangezogen werden sollte, um die Erkennung zu verbessern. Des Weiteren sind Ambiguitäten insbesondere bzgl. der Formate *h-mm* und *hhmm* zu erwarten. Erstgenanntes bietet Verwechslungspotential mit Raumnummern oder Bestandteilen von Telefonnummern und vierstellige Zahlen können insbesondere auch Jahreszahlen darstellen. Eine Berücksichtigung des Kontextes ist somit unumgänglich für eine zuverlässige Erkennung.

2.3 Architektur der Erkennungsmaschine

Dieser Abschnitt ist nicht öffentlich.

2.4 Anforderungen

An die Zeiterkennungsmaschine wird eine Reihe von Anforderungen gestellt. Neben einer möglichst korrekten und vollständigen Erkennung und Normalisierung ist eine schonende Ressourcennutzung von entscheidender Bedeutung, da die Maschine insbesondere auf mobilen Geräten, wie Smart-Phones und Feature-Phones[11] lauffähig sein soll. Mobile Geräte zeichnen sich durch eine geringe Prozessorleistung aus und besitzen nur eingeschränkten Arbeitsspeicher. Zur Verdeutlichung sind in Tab. 4 verschiedene Geräteklassen vergleichend dargestellt. Es ist erkennbar, dass insbesondere Feature-Phones äußerst begrenzte Rechenkapazitäten aufweisen und dadurch 25- bis 50-mal langsamer als durchschnittliche PCs sind.

	PC	Notebook	Smart-Phone	Feature-Phone	Tablet-PC
Prozessor[12]	x86 4 - 8 Kerne mit 3 - 4 GHz	x86 2 - 4 Kerne mit 2 - 3 GHz	ARM 2 - 4 Kerne mit 0,8 - 1,5 GHz	ARM 1 Kern mit 0,5 - 1 GHz	ARM 2 - 4 Kerne mit 1 - 1,5 GHz
Arbeitsspeicher	8 - 16 GB	4 - 8 GB	256 MB - 2 GB	128 MB	256 MB - 2 GB
Massenspeicher	1 - 4 TB	500 GB - 1 TB	8 - 32 GB	500 MB - 2 GB	8 - 32 GB

Tab. 3: Übersicht über Leistungsmerkmale verschiedener Geräteklassen

Trotz dieser Einschränkungen müssen die Ergebnisse – entsprechend den Anforderungen an eine mobile Anwendung – zügig geliefert werden. Als Maßgabe sind hierfür maximal zwei Sekunden für eine E-Mail mit 500 Wörtern vorgesehen. Bestehende Zeiterkennungsmaschinen würden indessen schon den gesamten verfügbaren Speicher eines Feature-Phones beanspruchen und auf diesen dennoch mehr Zeit benötigen [DM09]. Eine behutsame Verwendung von Ressourcen ist auch aus der

[11] Ein Feature-Phone ist ein Handy, welches deutlich schlechtere Leistungsmerkmalen als ein Smart-Phone aufweist. Dies kann sowohl auf hardwaretechnische Unterschiede, wie ein leistungsärmerer Prozessor oder eine fehlende GPS-Einheit, oder auf eine proprietäre Software zurückzuführen sein. Ein Feature-Phone ist daher in der Regel deutlich günstiger als ein Smart-Phone.

[12] ARM Prozessoren sind bei gleicher Taktfrequenz deutlich langsamer als x86-Prozessoren.

Perspektive des Energieverbrauchs sinnvoll, da vor allem eine überschwängliche Prozessor- oder RAM[13]-Nutzung den Akku eines mobilen Geräts zu sehr belastet.

Eine weitere Anforderung an die Zeiterkennungsmaschine ist die Gewährleistung der weitgehenden Sprachunabhängigkeit. Nur dadurch ist eine effiziente Skalierung auf viele Sprachen möglich. Die einzige sprachspezifische Komponente ist die Konfigurationsdatei für die jeweilig zu unterstützende Sprache. Eine darüberhinausgehende Verwendung von Sprachspezifika ist nicht vorgesehen. Dies schränkt insbesondere die Möglichkeit zur Bestimmung und die anschließende Verwendung der Zeitform von Verben stark ein, welche in Abschnitt 2.2.2 als hilfreich identifiziert wurde und den Kernteil von temporalsemantischen Modellen bildet.

2.5 Zusammenfassung

Dieser Abschnitt ist nicht öffentlich.

[13] von Englisch: random access memory (= Arbeitsspeicher)

3 Strategien zur Verbesserung

Dieses Kapitel ist nicht öffentlich.

4 Evaluation

Die im Kapitel 3 erarbeiteten Strategien zur Verbesserung sollen in diesem Kapitel evaluiert werden. Zu diesem Zweck werden in den Abschnitten 4.1 und 4.2 Maße zur Bewertung der Ergebnisse eingeführt und die Testdaten vorgestellt. Anschließend folgt in Abschnitt 4.3 die Darlegung der Evaluationsmethodik. Für die Testdaten wird in Abschnitt 4.4 die Fähigkeit der Maschine hinsichtlich Zeiterkennung und -normalisierung untersucht und in Abschnitt 4.5 häufige Fehler und deren Ursachen diskutiert. Zum Schluss wird im letzten Abschnitt 4.6 noch kurz auf das Laufzeitverhalten und den Speicherverbrauch eingegangen.

4.1 Evaluationsmaße

Das Evaluieren der Maschine bzw. des Verfahrens zur Zeiterkennung und -normalisierung setzt entsprechende Evaluationsmaße voraus. Bekannte und etablierte Maße zur Bewertung eines Verfahrens sind Precision, Recall und F-Measure.

Ein Verfahren erzeugt Antworten bzgl. einer Eingabe. Eine solche Antwort kann entweder korrekt oder inkorrekt sein. Sei E eine Eingabe. Dann bezeichne N die Anzahl aller möglichen korrekten Antworten bzgl. E, A die Anzahl der durch das Verfahren erzeugten Antworten bzgl. E, und T die Anzahl der davon korrekten Antworten.

Precision ist ein Maß, welches den Anteil der korrekt gegebenen Antworten des betrachteten Verfahrens bzgl. aller Antworten des Verfahrens angibt. Es gilt:

$$Precision = \frac{T}{A}$$

Recall ist ein Maß, welches den Anteil der korrekt gegebenen Antworten des betrachteten Verfahrens bzgl. aller möglichen korrekten Antworten angibt. Es gilt:

$$Recall = \frac{T}{N}$$

Sowohl *Precision* als auch *Recall* nehmen stets eine reelle Zahl zwischen 0 und 1 als Wert an. Im Idealfall haben beide den Wert 1, was bedeutet, dass jede gegebene Antwort korrekt ist und jede mögliche korrekte Antwort gegeben wurde. Normalerweise

schließt ein hoher Wert für eines der beiden Maße einen hohen Wert für das jeweils andere aus. Aufgrund von diesem Zusammenhang, wird ein Maß benötigt, das mit einem Wert die Gesamtqualität eines Verfahrens ausdrücken kann. Dies wird beispielsweise durch das F-Measure erreicht, welches das harmonische Mittel von Precision und Recall darstellt und somit gleichermaßen von beiden Werten abhängt. Es gilt:

$$F\text{-}Measure = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall}$$

Auch das F-Measure nimmt als Wert stets eine reelle Zahl zwischen 0 und 1 an.

4.2 Evaluationsdaten

Sprache	Datenquelle	Anzahl Texte	Texttyp[14]	Anzahl Zeitausdrücke
Deutsch	ExB-Gruppe	149	vorw. narrativ	919
Englisch	ExB-Gruppe	153	vorw. narrativ	1.151
	TimeBank 1.2[15]	183	informativ	1.414
	ACE 2005 Training Corpus[16]	599	informativ / narrativ	4.446
Französisch	ExB-Gruppe	9	vorw. narrativ	183
	Fr-TimeBank 1.0[17]	109	informativ	499
Spanisch	ExB-Gruppe	34	vorw. narrativ	325
Portugiesisch	ExB-Gruppe	29	vorw. narrativ	200
Koreanisch	ExB-Gruppe	55	vorw. narrativ	267
Chinesisch	ExB-Gruppe	29	vorw. narrativ	204
Finnisch	ExB-Gruppe	4	vorw. narrativ	64
Russisch	ExB-Gruppe	4	vorw. narrativ	92
Ukrainisch	ExB-Gruppe	3	vorw. narrativ	102

Tab. 4: Überblick über Umfang der Evaluationsdaten

[14] Die Klassifizierung ist [LNP04] entnommen.

[15] http://www.ldc.upenn.edu/Catalog/catalogEntry.jsp?catalogId=LDC2006T08

[16] http://www.ldc.upenn.edu/Catalog/catalogEntry.jsp?catalogId=LDC2006T06

[17] https://gforge.inria.fr/projects/fr-timebank/

Die Evaluationsdaten bestehen vollumfänglich aus manuell annotierten Texten und liegen für mehrere Sprachen vor. Sie stellen im weiteren Verlauf der Evaluation den sogenannten Gold-Standard dar. Zu den Evaluierungsdaten gehören einerseits von der ExB-Gruppe ausgewählte und anschließend annotierte Texte und andererseits allgemein bzw. kommerziell verfügbare Korpora, bei denen – unter anderem – Zeitausdrücke annotiert sind. Eine Übersicht über die Anzahl der Texte bzw. der enthaltenen temporalen Ausdrücke befindet sich in Tab. 4. Es ist ersichtlich, dass für ausgewählte Sprachen ein umfangreiches Evaluationsmaterial vorliegt: Deutsch, Englisch, Französisch. Für die anderen Sprachen liegen teilweise nur unzureichende Daten vor, so dass darauf aufbauende Evaluationsergebnisse kritisch begutachtet werden müssen – dies gilt insbesondere für Finnisch, Russisch und Ukrainisch.

4.3 Evaluationsmethodik

Als Basis für die Evaluation dient hauptsächlich eine Gegenüberstellung der ursprünglichen Implementierung und der – mit Hilfe der in Kapitel 3 erarbeiteten Strategien – verbesserten Maschine. Hierzu werden die beiden Maschinen mit denselben Texten und Sprachen getestet und deren Ergebnisse bzgl. des Gold-Standards verglichen.

Da die Maschine zwei Aufgaben erfüllt: Zeiterkennung und Zeitnormalisierung, ist es naheliegend diese beiden Aufgaben getrennt voneinander zu evaluieren. Dieser Ansatz wird auch konsequent in der Literatur verfolgt [CPS+10, DM09, LPS+06]. Um dem Umstand gerecht zu werden, dass die Normalisierung hochgradig von der Zeiterkennung abhängig ist, denn nur zuvor erkannte Zeitausdrücke können normalisiert werden, wird die Normalisierung nur auf den zuvor korrekt erkannten Zeitausdrücken untersucht. Fälschlicherweise erkannte sowie zuvor nicht erkannte Zeitausdrücke bleiben somit unberücksichtigt bei der Bewertung der Normalisierungsfähigkeit.

In den folgenden Abschnitten wird auf weitere Teilaspekte der Evaluationsmethodik eingegangen. Zunächst folgt eine Betrachtung der Match-Strategie. Diese steuert maßgeblich, ob eine gegebene Antwort korrekt ist oder nicht.

4.3.1 Fuzzy-Match-Strategie

Es wird keine exakte Übereinstimmung mit dem Gold-Standard gefordert, sondern es sind gewisse Abweichungen erlaubt. Dieses Vorgehen wird Fuzzy-Match-Strategie

genannt. Die Erkennung eines temporalen Ausdrucks wird genau dann als korrekt angesehen, wenn der erkannte Ausdruck mit genau einem Zeitausdruck im Gold-Standard (kurz: Gold-Annotation) überlappt. Falls es mehr als einen erkannten Ausdruck gibt, welcher sich mit dem Zeitausdruck aus dem Gold-Standard überschneidet, so wird nur eine der Antworten der Erkennungsmaschine als korrekt angesehen. Alle anderen Antworten werden als Erkennungsfehler behandelt.

Die Normalisierung wird ähnlich behandelt: Der bzw. die normalisierten Zeitpunkte müssen nicht exakt mit dem Gold-Standard übereinstimmen. Sie dürfen jedoch gewisse Abweichungen nicht überschreiten. Hierfür wird die Länge des Zeitintervalls der Gold-Annotation herangezogen. Sofern die Differenzen zwischen den ermittelten Zeitpunkten und denen der Gold-Annotation *zusammen* nur einen festgelegten Anteil des vorgegeben Intervalls aufweisen, wird die Normalisierung als korrekt betrachtet. Seien t_1 und t_2 der Start- bzw. der Endzeitpunkt des erkannten Zeitausdrucks und g_1 und g_2 die entsprechenden Zeitpunkte im Gold-Standard. Es muss dann folgendes gelten:

$$|t_1 - g_1| + |t_2 - g_2| \leq \alpha \cdot (g_2 - g_1)$$

Für α wird im weiteren Verlauf ein Wert von 0,5 angenommen. Zur Verdeutlichung der Fuzzy-Match-Strategie folgen in Tab. 5 einige Beispiele. Dabei wird auch auf die entkoppelte Evaluation von Erkennung und Normalisierung Bezug genommen. Der Eingabetext lautet: „Wir treffen uns heute Abend." Hierzu sei angenommen, dass *heute* der 21.09.2012 ist. Der Gold-Standard entspricht dann:

- „heute Abend"
- Startzeitpunkt: 21.09.2012 18:00:00
- Endzeitpunkt: 21.09.2012 22:00:00

Erkennung	Normalisierung	Bemerkung
„heute Abend“	21.09.2012 18:00:00 - 21.09.2012 22:00:00	korrekte Erkennung korrekte Normalisierung
„Abend“	21.09.2012 18:00:00 - 21.09.2012 22:00:00	korrekte Erkennung korrekte Normalisierung
„heute“ „Abend“	21.09.2012 00:00:00 - 21.09.2012 23:59:59 21.09.2012 17:30:00 - 21.09.2012 21:30:00	teilw. fehlerhafte Erkennung korrekte Normalisierung
---	---	fehlerhafte Erkennung Norm. bleibt unberücksichtigt
„Wir“	21.09.2012 12:56:43	fehlerhafte Erkennung Norm. bleibt unberücksichtigt
„uns heute“	21.09.2012 00:00:00 - 21.09.2012 23:59:59	korrekte Erkennung fehlerhafte Normalisierung
„heute Abend“	21.09.2012 17:00:00 - 21.09.2012 20:00:00	korrekte Erkennung fehlerhafte Normalisierung

Tab. 5: Beispiele zur verwendeten Fuzzy-Match-Strategie

Das erste Beispiel entspricht genau dem Gold-Standard und ist daher korrekt. Im zweiten Beispiel ist zwar nur „Abend“ erkannt, aber da sich dieser Text mit dem des Gold-Standards überschneidet, gilt auch hier die Erkennung als korrekt. Die Erkennung im dritten Beispiel ist nur teilweise richtig, da fälschlicherweise zwei Ausdrücke erkannt werden, demgegenüber ist die Normalisierung korrekt, da der normalisierte Wert des fälschlich erkannten Ausdrucks „heute“ unberücksichtigt bleibt und die Normalisierung von „Abend“ nur eine geringe Abweichung zum Gold-Standard aufweist. Die Länge des Zeitintervalls des Gold-Standards beträgt vier Stunden. Das bedeutet, dass der Start- und der Endzeitpunkt zusammen insgesamt um zwei Stunden abweichen dürfen. Da beide Zeitpunkte jeweils 30 Minuten vom entsprechenden Zeitpunkt des Gold-Standards abweichen, ist der Unterschied hinreichend gering. In den Fällen vier und fünf ist die Erkennung fehlgeschlagen. Jedoch haben auch hier die ggfs. ermittelten normalisierten Werte keine Relevanz für die Evaluation. Die Erkennung bei den letzten beiden Beispielen ist hingegen wieder korrekt, so dass die normalisierten Zeiten für die Evaluation herangezogen werden. In beiden Fällen ist die Normalisierung aber inkorrekt. Der Startzeitpunkt im letzten Beispiel weicht zwar nur eine Stunde vom Startzeitpunkt des

Gold-Standards ab, allerdings weist der Endzeitpunkt einen Abstand von zwei Stunden auf. Die Summe dieser Abweichungen liegt dadurch mit drei Stunden jenseits der Toleranzgrenze.

Die hier verwendete Match-Strategie soll das Nutzererlebnis widerspiegeln: Diesem ist es in der Regel unwichtig, ob der entsprechende Textbestandteil exakt erkannt wurde – solange der normalisierte Wert des Zeitausdrucks plausibel ist.

4.3.2 Untere und obere Schranke

Um die Fähigkeit der Maschine besser einschätzen zu können, werden eine untere Schranke und – soweit verfügbar – eine obere Schranke angegeben. Dabei handelt es sich um Ergebniswerte, welche üblicherweise durch kein (automatisches) Verfahren unter- bzw. überschritten werden können. Für die Ermittlung der unteren Schranke wird ein Baseline-Algorithmus eingesetzt. Dieser arbeitet vollständig sprachunabhängig und kann somit für jede Sprache eingesetzt werden. Es werden dabei nur vollständig spezifizierte Datums- und Uhrzeitangaben, bestehend aus arabischen Ziffern und verschiedenen Trennzeichen, sowie Jahresangaben erkannt.

Um eine obere Schranke zu ermitteln, wird das sogenannte Inter-Annotator-Agreement (kurz: IAA) herangezogen. Dieses ist ein Maß dafür, wie einig sich zwei oder mehr unterschiedliche Annotatoren sind. Hierzu werden deren annotierte Texte paarweise miteinander verglichen. Der Schluss, dass es sich dabei tatsächlich um eine obere Schranke für einen automatischen Algorithmus handelt, fällt ziemlich leicht, denn wenn bereits zwei Menschen zu unterschiedlichen Ergebnissen kommen, so kann eine Maschine nur schwer zu einem allgemein anerkannten Ergebnis kommen. Beachtet werden muss hier, dass die innerhalb der ExB-Gruppe erzeugten Evaluierungsdaten ohne umfangreiche Richtlinien[18] entstanden sind – im Gegensatz zu denen der anderen Quellen [FGM+05, GKL+06]. Entsprechend ist das IAA vergleichsweise gering. Das IAA liegt für die Sprachen Deutsch und Englisch vor.

[18] Richtlinien unterstützen die Annotatoren, indem sie vorgeben, welche Ausdrücke annotiert werden sollen und welche nicht. Außerdem geben sie Hinweise, worauf im Detail geachtet werden muss.

4.3.3 Micro- und macro-averaging

Im Rahmen der Evaluation wird eine Vielzahl von Sprachen evaluiert. Dies hat zur Folge, dass eine Mittelwertbildung notwendig ist, um eine globale Aussage bzgl. der Ergebnisse der Maschine treffen zu können. Es gibt zwei übliche Verfahren: micro-averaging und macro-averaging. Sei k die Anzahl verschiedener Sprachen. Analog zu Abschnitt 4.1 seien N_i, A_i und T_i jeweils die Anzahl aller möglichen korrekten Antworten, die Anzahl der von der Maschine erzeugten Antworten und die davon korrekten Antworten. $Precision_i$ und $Recall_i$ bezeichnen dann die jeweiligen Werte für Precision und Recall der i-ten Sprache.

Beim macro-averaging werden die Werte für Precision bzw. Recall der einzelnen Sprachen addiert und durch die Anzahl der Sprachen geteilt. Es gilt:

$$Precision_{macro} = \frac{1}{k} \cdot \sum_{i=1}^{k} Precision_i = \frac{1}{k} \cdot \sum_{i=1}^{k} \frac{T_i}{A_i}$$

$$Recall_{macro} = \frac{1}{k} \cdot \sum_{i=1}^{k} Recall_i = \frac{1}{k} \cdot \sum_{i=1}^{k} \frac{T_i}{N_i}$$

Beim micro-averaging wird kein Unterschied zwischen den einzelnen Sprachen gemacht. Stattdessen werden alle Antworten – sowohl die korrekten als auch die von der Maschine abgegebenen – addiert und daraus Precision bzw. Recall berechnet. Es gilt:

$$Precision_{micro} = \frac{\sum_{i=1}^{k} T_i}{\sum_{i=1}^{k} A_i}$$

$$Recall_{micro} = \frac{\sum_{i=1}^{k} T_i}{\sum_{i=1}^{k} N_i}$$

Macro-averaging bietet sich vor allem dann an, wenn stark unterschiedliche große Testmengen vorliegen, da es beim micro-averaging naturgemäß zu einer Überbewertung der großen Testmengen kommt und entsprechend zu einer Unterbewertung der kleinen Testmengen. Für diese Evaluation ist es daher naheliegend auf micro-averaging zu verzichten und das Ergebnis über alle Sprachen per macro-averaging zu bestimmen.

Die Entscheidung für eine dieser beiden Methoden ist an einer weiteren Stelle von Relevanz. Wie in Abschnitt 4.2 angegeben, werden auch Sprachen evaluiert, bei denen mehr

als eine Datenquelle herangezogen wird. Die Anzahl der Testfälle aus den jeweiligen Quellen variiert sehr stark, so dass auch hier macro-averaging verwendet wird. Die von der ExB-Gruppe erzeugten Evaluationsdaten hätten sonst nur eine geringe Auswirkung auf das Evaluationsergebnis, obwohl es sich um bessere[19] Daten handelt.

4.4 Ergebnisse

Wie in Abb. 13 zu erkennen ist, erreicht der Baseline-Algorithmus (*BL*) zwar einen sehr hohen Wert für die Precision bzgl. der Erkennung von temporalen Ausdrücken, jedoch ist der Recall aufgrund des eingeschränkten Fokus gering, was schließlich auch in einem geringen F-Measure resultiert. Im Vergleich dazu sind sowohl die Ausgangsmaschine (*A*) als auch die verbesserte Maschine (*V*) signifikant besser. Ein ähnliches Bild zeigt sich für die Normalisierung. Hier ist die mäßige Leistung von *BL* und *A* darauf zurückzuführen, dass keine Berücksichtigung des Datumsformats stattfindet.

Auch im direkten Vergleich zwischen *A* und *V* ergibt sich eine deutlich bessere Performance für *V*. *A* erreicht bei der Erkennung ein F-Measure von 0,63 und bei der Normalisierung 0,57. Somit liegt das F-Measure von *V* 19% bzw. 21% über dem von *A*.

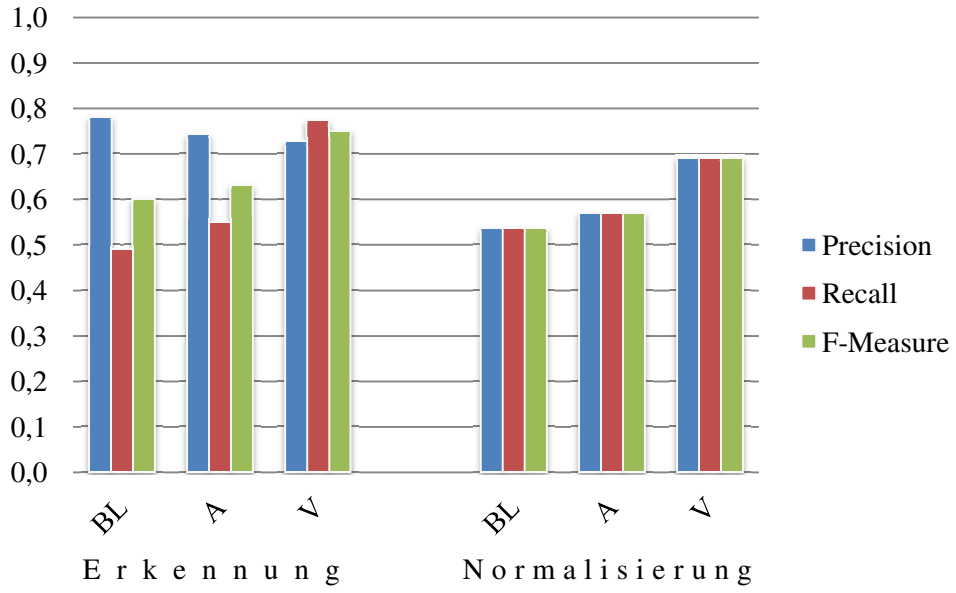

Abb. 4: Ergebnis der Evaluation über alle Sprachen

In den folgenden Abschnitten werden die Evaluationsergebnisse für Deutsch und Englisch detaillierter betrachtet und darin zusätzlich auf das Inter-Annotator-Agreement

[19] Besser im Sinne von näher am angedachten Anwendungsfall: Natürlichsprachige Kommunikation zwischen Nutzern.

eingegangen. Im Anschluss daran folgt eine nähere Betrachtung des Einflusses ausgewählter Verbesserungsstrategien. Hierfür werden in verschiedenen Evaluationsszenarios einzelne Aspekte der verbesserten Maschine ausgeblendet. Schließlich wird in Abschnitt 4.4.4 ein Überblick über die Ergebnisse aller evaluierten Sprachen gegebenen.

4.4.1 Deutsch

Analog zum vorhergehenden Abschnitt stehen in Abb. 14, *A* für Ausgangstand, *V* für verbesserte Maschine und *BL* für Baseline-Verfahren. Zusätzlich wird das IAA betrachtet (*IAA*). Aufgrund der Symmetrie beim Vergleichen der Texte gilt für *IAA* – wie bei der Normalisierung – dass Precision und Recall stets den gleichen Wert aufweisen. In dessen Bestimmung sind insgesamt 94 Texte von sieben Annotatoren eingeflossen.

Obwohl das IAA eine obere Schranke vorgeben soll, ist *V* offensichtlich besser als *IAA*. Dies gilt insbesondere für die Normalisierung. Die Gründe hierfür sind vielfältig. Ein Hauptgrund ist die Fuzzy-Match-Strategie, denn dadurch ist es möglich, dass obwohl zwei Annotatoren eine Zeitspanne unterschiedlich interpretieren, die von der Maschine erzeugte Interpretation dennoch mit beiden übereinstimmt. Darüber hinaus hat es, wie bereits in Abschnitt 4.3.2 dargelegt, zum Annotationsprozess kaum Vorgaben für die Annotatoren gegeben.

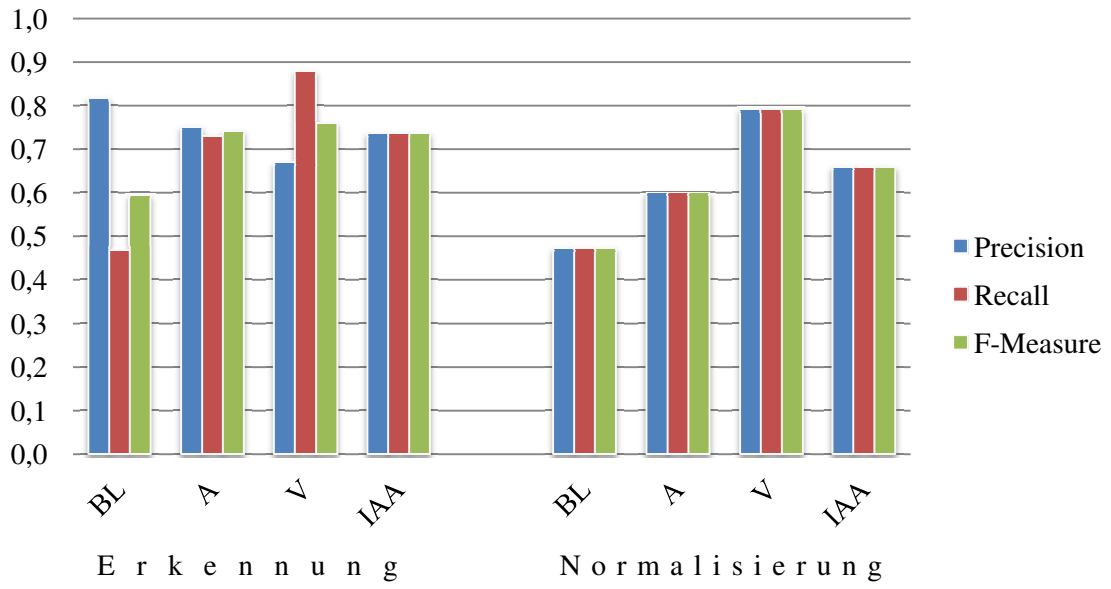

Abb. 5: Ergebnis der Evaluation für Deutsch

Auffällig ist weiterhin, dass bereits der Ausgangsstand einen überdurchschnittlichen Wert für den Recall bei der Zeiterkennung aufweist und die Precision – ebenfalls bzgl.

der Zeiterkennung – bei *V* schlechter ist als bei *A*. Die Ursache für Letzteres ist vor allem ein fehlerhafter Gold-Standard, siehe Abschnitt 4.5. Der hohe Recall-Wert bei *A* ist darauf zurückzuführen, dass die deutsche Konfiguration bereits zu Beginn sehr umfangreich gewesen ist und dabei auch zahlreiche morphologische Variationen berücksichtigt worden sind. Dennoch haben die Erweiterung der Regelmenge und der Konfiguration eine weitere Steigerung des Recall bei *V* ermöglicht.

4.4.2 Englisch

Das Baseline-Verfahren ist im Englischen hinsichtlich der Erkennung von Zeitausdrücken deutlich schlechter als im Deutschen, denn der Wert für den Recall beträgt laut Abb. 15 gerade einmal knapp 28%. Hintergrund ist das umfangreiche Evaluationsmaterial und dessen vorherrschender Texttyp: informativ. Dieser ist mehr durch relative Zeitangaben wie zum Beispiel Wochentage geprägt als narrativer Text. Auch die gute Normalisierung von *BL* lässt sich somit erklären: Das standardmäßig angenommene Datumsformat MM/DD/YY trifft auf Datumsangaben in englischen informativen Texten überproportional oft zu.

Der im Vergleich zur deutschen Evaluation niedrige Recall von *A* und *V* bei der Erkennung liegt ebenso an den zusätzlichen Datenquellen, denn wenn nur die Evaluationsdaten der ExB-Gruppe betrachtet werden, liegt der Recall-Wert auf gleicher Höhe mit dem der deutschen Evaluation. Wie bereits angedeutet ist die Breite an verschiedenartigen Zeitausdrücken im ACE Korpus sowie in der TimeBank deutlich ausgeprägter. Insofern ist es zunächst nicht nachvollziehbar, dass die Precision bzgl. der Erkennung trotzdem deutlich höher ist als im Deutschen. Dies lässt sich nicht allein auf den Texttyp zurückführen. Stattdessen kommen hier andere Ursachen in Betracht: Zum einen die umfangreichen Richtlinien bei den externen Datenquellen und zum anderen ist die englische Sprache und deren Satzbau bzw. Wortstellung einfacher.

IAA basiert auf 59 Texten von sechs Annotatoren. Es sind somit vergleichsweise viele Annotatoren an der Bestimmung des englischen IAA beteiligt, die jeweils aber nur wenige Texte annotiert haben. Durch den eingeschränkten Fokus ist die statistische Relevanz begrenzt.

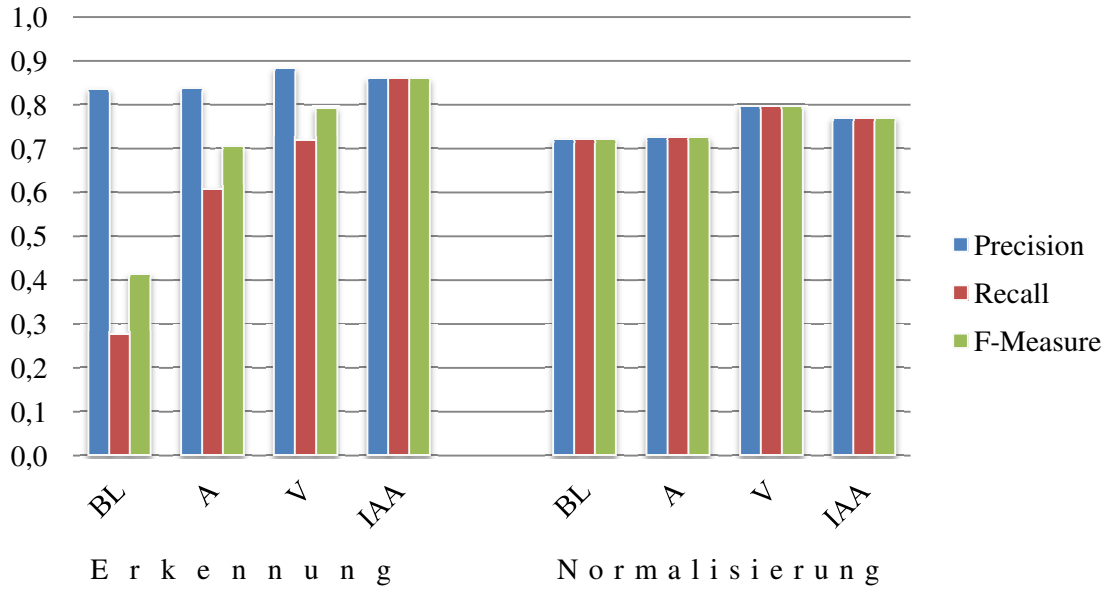

Abb. 6: Ergebnis der Evaluation für Englisch

4.4.3 Einfluss einzelner Verbesserungsstrategien

Dieser Abschnitt ist nicht öffentlich.

4.4.4 Übersicht

Tab. 6 zeigt eine nach Sprache aufgeschlüsselte Übersicht über die Ergebnisse der Evaluation. Zudem wird die relative Änderung der gemachten Fehler der Zeiterkennungsmaschine bzgl. falscher und fehlender Antworten angegeben. Rot steht dabei für eine Verschlechterung und Grün für eine Verbesserung. Der Tabelle sind somit für nahezu alle evaluierten Sprachen eine Verringerung der Fehler zu entnehmen. Dies gilt insbesondere für die Anzahl fehlender Antworten. Eine Ausnahme bildet Finnisch, bei der objektiv weder bzgl. der Erkennung noch bzgl. der Normalisierung eine Verbesserung eintritt. Hier sei noch einmal betont, dass für Finnisch nur außerordentlich wenig Evaluationsmaterial vorliegt, so dass diese Verschlechterung nicht signifikant ist. Bei der Beurteilung der Normalisierung ist weiterhin zu beachten, dass die Anzahl potentiell falscher Antworten mit der Zunahme des Recall bei der Zeiterkennung steigt, da die Normalisierung, wie in Abschnitt 4.3 dargelegt, nur für korrekt erkannte Zeitausdrücke untersucht wird. Aus diesem Grund sind die Zahlen zur Änderung der Anzahl falscher Antworten bei der Normalisierung um diesen Effekt bereinigt.

Sprache	Erkennung					Normalisierung	
	Precision	Recall	F-Measure	Änderung Anzahl falscher Antworten	Änderung Anzahl fehlender Antworten	P/R/F	Änderung Anzahl falscher Antworten
Deutsch	0,67	0,88	0,76	33,3%	55,5%	0,79	47,8%
Englisch	0,88	0,72	0,79	28,4%	28,2%	0,80	25,8%
Französisch	0,73	0,78	0,75	25,4%	34,9%	0,78	45,9%
Spanisch	0,72	0,77	0,74	43,3%	40,3%	0,61	7,8%
Portugiesisch	0,84	0,76	0,80	15,7%	48,4%	0,66	4,5%
Koreanisch	0,67	0,49	0,56	5,0%	30,1%	0,59	11,7%
Chinesisch	0,86	0,91	0,89	37,7%	88,7%	0,77	59,8%
Finnisch	0,64	0,80	0,71	121,2%	0,0%	0,76	1,9%
Russisch	0,67	0,80	0,73	2,6%	70,0%	0,69	54,0%
Ukrainisch	0,60	0,83	0,70	0,9%	32,0%	0,46	14,0%
Gesamt (macro-average)	0,73	0,77	0,75	9,9%	42,8%	0,69	23,7%

Tab. 6: Übersicht über Ergebnisse der Evaluation

4.5 Fehleranalyse

Abschließend folgt eine Betrachtung der Fehler, welche die Maschine sowohl beim Erkennen als auch beim Normalisieren von temporalen Ausdrücken macht. Dies soll das Verständnis der Evaluationsergebnisse aus dem vorangegangen Abschnitt weiter vertiefen. Zu diesem Zweck werden sämtliche Fehler der Maschine bzgl. des Gold-Standards für Deutsch untersucht und nach der Ursache kategorisiert. Die Fehleranalyse wird in drei unabhängigen Teilen durchgeführt:

- Falsche Antworten bei der Erkennung
- Fehlende Antworten bei der Erkennung
- Falsche Antworten bei der Normalisierung

Die zusätzliche Aufteilung der Erkennungsfehler ist notwendig, da sich die Ursachen für fälschlicherweise als Zeitausdruck erkannte Textbestandteile deutlich von denen

nicht erkannter Zeitausdrücke unterscheiden. In Abb. 17 sind zunächst alle – im Rahmen der Zeiterkennung – als inkorrekt gegebenen Antworten und deren Ursachen aufgeführt. Insgesamt gibt es acht verschiedene Ursachen. In etwa einem Drittel der Fälle handelt es sich im Grunde um gar keinen Fehler der Maschine, sondern ein fehlerhafter Gold-Standard ist die Ursache für den vermeintlichen Fehler. Dieselbe Beobachtung wird auch in [DM08] gemacht. Dies bestätigt auch die in Abschnitt 1.3 gemachte Aussage, dass es sich bei der Zeiterkennung um eine äußerst anspruchsvolle Aufgabe handelt. Die zweitgrößte Ursache für Fehler ist fehlendes Kontextwissen. Das bedeutet, dass unter Berücksichtigung des Kontextes, in dem der jeweilige Zeitausdruck steht und ggfs. unter Zuhilfenahme von Weltwissen, eine falsche Erkennung verhindert werden könnte. Konkret sind sehr häufig Versionsnummern von Software oder Standards der Grund für diese Fehler, denn Versionsnummern sind (im Deutschen) nur schwer von Datumsangaben zu unterscheiden („HTML 4.1“ vs. „der 4.1 ist verregnet“). Das liegt daran, dass im Deutschen normalerweise als Dezimaltrennzeichen ein Komma verwendet wird. Die dritthäufigste Ursache für Fehler dieser Kategorie sind figurative Ausdrücke. Dabei handelt es sich zwar grundsätzlich auch um Zeitausdrücke, jedoch lassen sich diese durch ihre inhärente Ungenauigkeit nur sehr schwer normalisieren und werden daher von den Annotatoren häufig gar nicht annotiert. Ein Beispiel lautet: „Früher war alles schöner als *heute*.“ Heute bezieht sich dabei nicht auf den aktuellen Tag, sondern auf die heutige Zeit. Die restlichen Fehler verteilen sich auf diverse andere Ursachen.

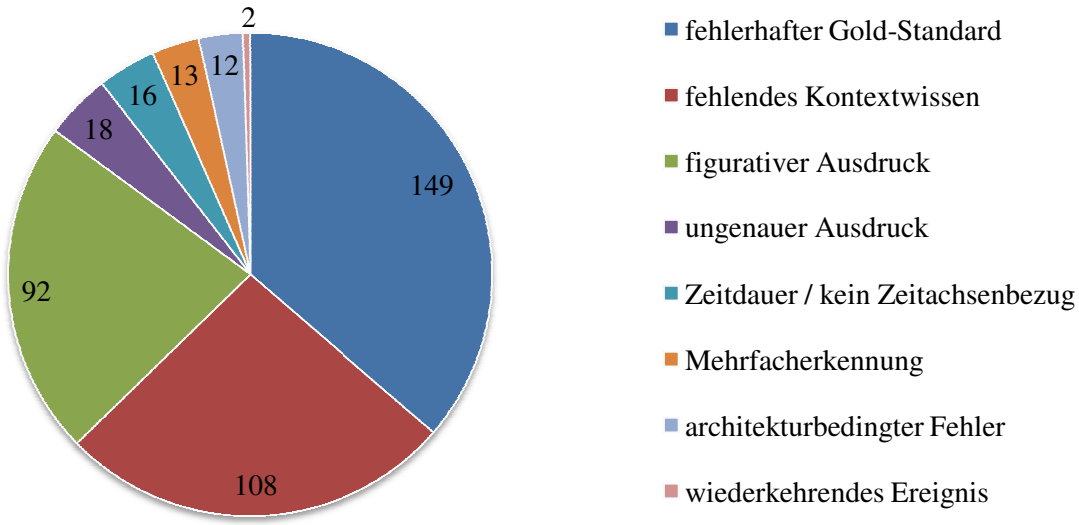

Abb. 7: Anzahl falscher Antworten bei der Zeiterkennung (nach Ursache)

Das Nichterkennen von temporalen Ausdrücken lässt sich ebenfalls auf acht verschiedene Ursachen zurückführen, siehe Abb. 18. Besonders auffällig ist zunächst, dass hier Annotator-Fehler kaum auftreten. Die Diskrepanz zwischen falscher Erkennung und fehlender Erkennung bzgl. der Annotator-Fehler ist leicht zu erklären: Ein Mensch übersieht sehr leicht einen Ausdruck, der eigentlich zu annotieren wäre, aber wenn er einen Textbestandteil als Zeitausdruck identifiziert und entsprechend annotiert, so ist diese Annotation fast immer auch korrekt.

Hauptgrund für das Scheitern der Zeiterkennung sind historische bzw. futuristische Jahresangaben. Zwar handelt es sich dabei stets um vierstellige Zahlen, diese sind allerdings nur in einem eingeschränkten Intervall zuverlässig als Jahreszahlen interpretierbar. Für Jahresangaben außerhalb dieses Intervalls muss der Kontext analysiert werden, um so beispielsweise einfache Mengenangaben („4500 Menschen kamen zur Kundgebung“) von echten Jahresangaben unterscheiden zu können. Die Nichterkennung von Ausdrücken mit fehlendem Zeitachsenbezug ist ebenso wie die Nichterkennung von ungenauen Ausdrücken und wiederkehrenden Ereignissen darauf zurückzuführen, dass sie durch das Speicherformat der Maschine nicht dargestellt werden können und daher auch nicht verarbeitet werden. In knapp 20% aller Fehlerfälle ist die Ursache eine fehlende Regel, die den jeweiligen Ausdruck erkennen kann. In einem Fall führte auch ein Schreibfehler im Eingabetext zu einem Fehler in der Erkennung.

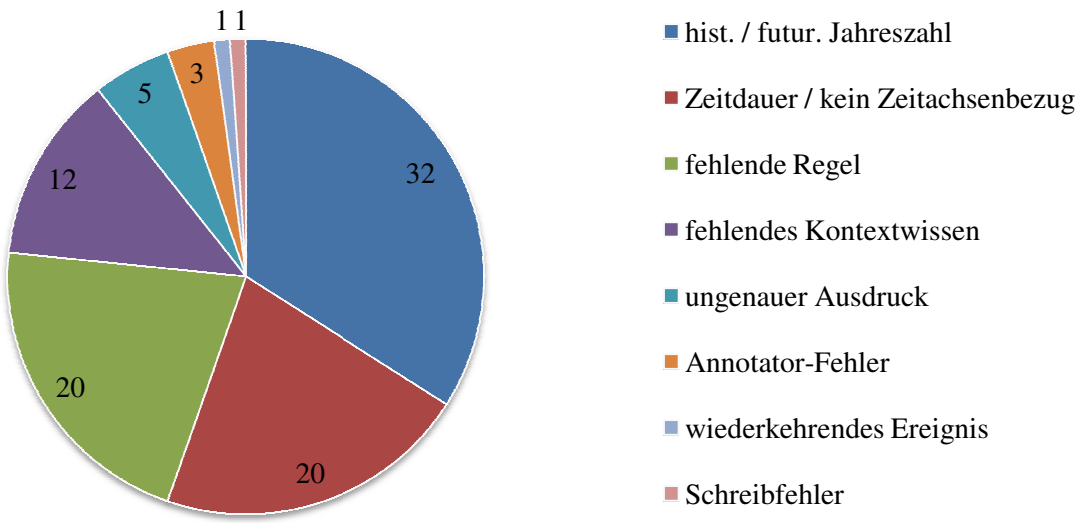

Abb. 8: Anzahl fehlender Antworten bei der Zeiterkennung (nach Ursache)

Bei der Normalisierung zeigt sich ein ähnliches Bild wie bei der Fehleranalyse falscher Antworten bzgl. der Zeiterkennung, siehe Abb. 19: Es dominieren Annotator-Fehler und

die unzureichende Verwendung von Kontextwissen. Eine weitere Ursache ist die zuvor nur partielle Erkennung eines Ausdrucks, so dass die Normalisierung – mangels Bereitstellung vollständiger Information – zwangsweise fehlschlägt. Dies passiert zum Beispiel bei der Wortgruppe „am Freitag, den 13.05. um 10-12 Uhr“, denn es wird nur „am Freitag, den 13.05.“ erkannt. Erwartungsgemäß führt die Nicht-Berücksichtigung der Zeitform zu einer Reihe von Fehlern und zwar hauptsächlich im Zusammenhang mit Wochentagen. Weiterhin wird beispielsweise „in der Nacht zu Donnerstag“ falsch interpretiert. Es werden zwar *Nacht* und *Donnerstag* korrekt erkannt, die Kombination der beiden Teilausdrücke führt jedoch zu der gleichen Repräsentation, wie „Donnerstagnacht“. Diese Art von Fehler lässt sich auf eine fehlerhafte Architektur zurückführen. Nicht zuletzt sind auch wiederkehrende Ereignisse im Gold-Standard annotiert, so dass diese gleichzeitig Annotator-Fehler darstellen, denn solche Ereignisse können im gegenwärtigen Speicherformat gar nicht repräsentiert werden.

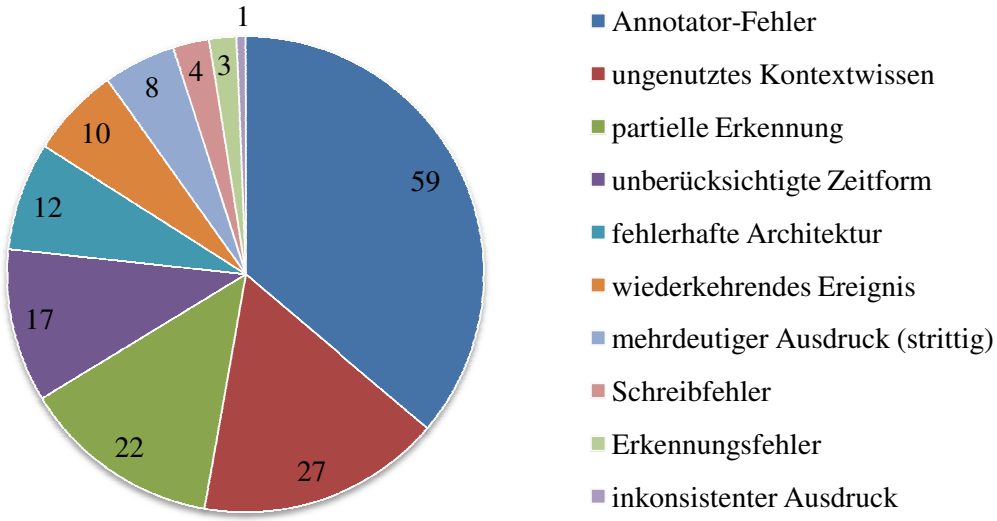

Abb. 9: Anzahl falscher Antworten bei der Normalisierung (nach Ursache)

4.6 Laufzeitverhalten und Speicherverbrauch

Dieses Kapitel hat bisher einen umfassenden Einblick in die Fähigkeit der Zeiterkennungsmaschine gegeben. An dieser Stelle soll nun eine kurze Betrachtung des Laufzeitverhaltens und des Speicherverbrauchs erfolgen. In Abb. 20 wird hierzu erneut die verbesserte Maschine (*V*) mit dem Ausgansstand (*A*) verglichen. Es ist erkennbar, dass bei beiden Maschinen mit steigender Länge des Textes vor allem die Laufzeit linear ansteigt. Der Speicherverbrauch von *V* ist verglichen mit *A* zwar gestiegen, gleichzeitig ist aber die Laufzeit zum Teil viermal geringer und das obwohl einige rechenintensive

Berechnungen, wie zum Beispiel die Kompositazerlegung, hinzugekommen sind. Dies ist auf umfassende Optimierungen zurückzuführen.

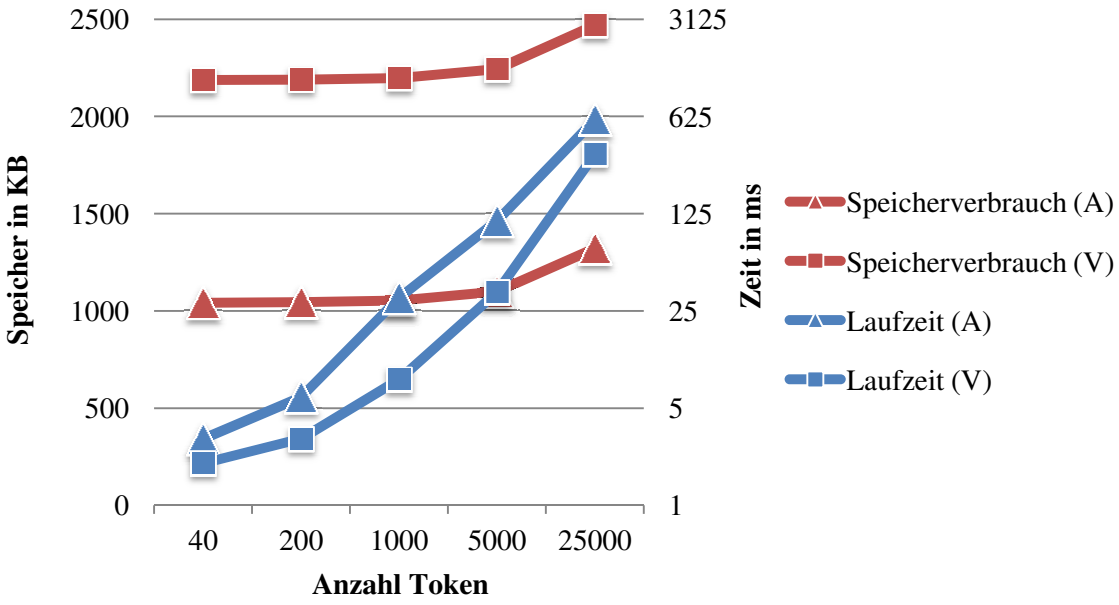

Abb. 10: Laufzeitverhalten und Speicherverbrauch

5 Schluss

In diesem finalen Kapitel werden die Ergebnisse der Arbeit kurz zusammengefasst und Weiterentwicklungsmöglichkeiten aufgezeigt.

5.1 Zusammenfassung

Zeiterkennung und -normalisierung stellt ein komplexes und herausforderndes Problem dar. Dies gilt insbesondere unter Berücksichtigung der Einschränkungen einer mobilen Plattform sowie unter Beachtung einer Mehrsprachigkeit. Selbst die manuelle Annotation von temporalen Ausdrücken ist nicht trivial, was sich in einem vergleichsweise niedrigen Inter-Annotator-Agreement manifestiert.

Es wurde eine ausführliche und systematische Analyse der Ausgangslage durchgeführt, um Ursachen zu identifizieren, die das Erkennen und das Normalisieren von Zeitangaben erschweren. Mit Hilfe daraus resultierender Erkenntnisse wurden zunächst Ansatzpunkte und später ausgereifte Strategien zur Verbesserung einer bestehenden Maschine entwickelt. Als wichtigste Strategien haben sich hierzu einerseits die Erweiterung der Regelmenge und andererseits die (semi-)automatische Unterstützung bei der Erstellung einer Konfigurationsdatei bewährt. Insbesondere durch den letztgenannten Punkt konnte eine Reduktion des Aufwands für die Erstellung einer solchen Datei und gleichzeitig eine Steigerung der Evaluationsergebnisse erreicht werden.

Es konnte gezeigt werden, dass ein regelbasierter Ansatz mit Fokus auf weitgehende Sprachunabhängigkeit vielversprechend und – trotz bisher manuell erstellter Regeln – eine effiziente Umsetzung möglich ist. Jede der vorgestellten Strategien wurde im Rahmen dieser Masterarbeit implementiert und in die bestehende Architektur der Zeiterkennungsmaschine der ExB-Gruppe integriert. Diesbezüglich konnte auch die eingangs geforderte weitgehende Sprachunabhängigkeit gewahrt bleiben und sowohl die Erkennungsrate als auch die Normalisierungsfähigkeit signifikant verbessert werden.

Darüber hinaus wurde mit Hilfe der im Zusammenhang mit der Erkennungs- und Normalisierungsmaschine erstellten sprachspezifischen Konfigurationen ein weiteres Verfahren umgesetzt ohne zusätzliche sprachspezifische Ressourcen generieren oder die ursprünglichen Daten anpassen zu müssen: Eine Maschine zum *Erzeugen* von

natürlichsprachigen Zeitausdrücken aus normalisierten Zeiten unter Berücksichtigung der aktuellen Zeit. Die relative Länge des repräsentierten Zeitintervalls sowie die maximale Anzahl der Zeichen des Ausdrucks können durch Parameter festgelegt werden. Die Details zu diesem Verfahren müssen leider unbetrachtet bleiben, da es den Rahmen dieser Arbeit sprengen würde.

5.2 Weiterentwicklungsmöglichkeiten

Durch systematisches Beseitigen der innerhalb der Fehleranalyse angesprochenen Fehlerursachen ist eine deutliche Steigerung in allen Bereichen zu erwarten und dies nicht nur für die konkret betrachtete Sprache Deutsch. Die Maschine zeigt allerdings nicht für alle Sprachen gleich gute Leistungsmerkmale. Insbesondere morphologisch komplexe Sprachen, wie etwa Finnisch, können nicht hinreichend gut verarbeitet werden. Weiterführende Abstraktion der morphologischen Vielfalt ist daher zielführend.

Trotz maschineller Unterstützung bei der Erstellung bzw. Erweiterung von Konfigurationsdateien, sind diese Prozesse weiterhin von manuellem Aufwand geprägt und bietet weiteres Potential zur Automatisierung. Eine Möglichkeit ist es, zu den einzelnen Konzepten weitere Wörter zu generieren, die synonym oder ähnlich zu darin bereits vorhandenen Wörtern sind und somit das gleiche Konzept repräsentieren. In diesem Bereich hat bereits umfangreiche Forschung stattgefunden. Ein Überblick wird zum Beispiel in [B07] gegeben. Erste Versuche sind indessen jedoch daran gescheitert, dass zu viele unpassende Wörter gelernt werden und dadurch die Precision der Zeiterkennungsmaschine in größerem Maße sinkt, als der Recall steigt.

Eine andere Weiterentwicklungsmöglichkeit ist die semi-automatische Generierung von Regeln. Durch maschinelle Lernverfahren ist es möglich, Kandidaten für neue Regeln zu generieren und diese aufbereitet darzustellen, um im Anschluss daran geeignete Kandidaten auszuwählen. Eine weiterführende Automatisierung dieses Prozesses könnte dann sogar dazu genutzt werden, *sprachspezifische* Regeln zu lernen und zu verwenden, um so vor allem den Recall weiter zu steigern.

Auch werden die einzelnen Typen[20] von temporalen Ausdrücken unterschiedlich gut unterstützt. So ist beispielsweise das Erkennen von wiederkehrenden Ereignissen nicht

[20] Typ steht hier für die Gesamtheit an Ausprägungen der einzelnen Eigenschaften, siehe Abschnitt 2.1.

möglich. Der Grund hierfür ist, dass weitreichende Änderungen – sowohl am Datenformat als auch an der Maschine selbst – notwendig sind, um wiederkehrende Ereignisse abbilden zu können. Nicht zuletzt ist es auch zielführend ungenaue Zeitausdrücke und Zeitangaben ohne konkreten Zeitachsenbezug zu verarbeiten – zumindest, um diese gezielt ignorieren zu können. Gegenwärtig kommt es gelegentlich vor, dass ein solcher Ausdruck bzw. ein Teil davon fälschlicherweise als exakt bzw. als Zeitausdruck mit konkretem Zeitachsenbezug erkannt und interpretiert wird.

Kurzzusammenfassung

Digital gespeicherte Daten erfreuen sich einer stetig steigenden Verwendung. Insbesondere die computerbasierte Kommunikation über E-Mail, SMS, Messenger usw. hat klassische Kommunikationsmittel nahezu vollständig verdrängt. Einen Mehrwert aus diesen Daten zu generieren, ist sowohl im geschäftlichen als auch im privaten Bereich von entscheidender Bedeutung. Eine Möglichkeit den Nutzer zu unterstützen ist es, seine textuellen Daten umfassend zu analysieren und bestimmte Elemente hervorzuheben und ihm die Erstellung von Einträgen für Kalender, Adressbuch und dergleichen abzunehmen bzw. zumindest vorzubereiten. Eine weitere Möglichkeit stellt die semantische Suche in den Daten des Nutzers dar. Selbst mit Volltextsuche muss man bisher den genauen Wortlaut kennen, wenn man eine bestimmte Information sucht. Durch ein tiefgreifendes Verständnis für *Zeit* ist es nun aber möglich, über einen Zeitstrahl alle mit einem bestimmten Zeitpunkt oder einer Zeitspanne verknüpften Daten zu finden. Es existieren bereits viele Ansätze um Named Entity Recognition voll- bzw. semi-automatisch durchzuführen, aber insbesondere Verfahren, welche weitgehend sprachunabhängig arbeiten und sich somit leicht auf viele Sprachen skalieren lassen, sind kaum publiziert. Um ein solches Verfahren für natürlichsprachige Zeitausdrücke zu verbessern, werden in dieser Arbeit, basierend auf umfangreichen Analysen, Möglichkeiten vorgestellt. Es wird speziell eine Strategie entwickelt, die auf einem Verfahren des maschinellen Lernens beruht und so den manuellen Aufwand für die Unterstützung neuer Sprachen reduziert. Diese und weitere Strategien wurden implementiert und in die bestehende Architektur der Zeiterkennungsmaschine der ExB-Gruppe integriert.

Literaturverzeichnis

[ABG07] Alonso, O.; Baeza-Yates, R.; Gertz, M.: *On the Value of Temporal Information in Information Retrieval*. In: ACM SIGIR Forum 41(2). 2007

[ABS08] Allen, J.; de Beaumont, W.; Swift, M.: *Deep Semantic Analysis of Text*. Proc. of the Conference on Semantics in Systems for Text Processing (STEP) 2008

[ARR07] Ahn, D.; van Rantwijk, J.; de Rijke, M.: *A Cascaded Machine Learning Approach to Interpreting Temporal Expressions*. Proc. of the Annual Conference of the North American Chapter of the Association for Computational Linguistics - Human Language Technologies (NAACL-HLT) 2007

[AU10] Allen, J.; UzZaman, N.: *TRIPS and TRIOS System for TempEval-2: Extracting Temporal Information from Text*. Proc. of the 5th International Workshop on Semantic Evaluation 2010

[B02] Baldwin, J.: *Learning Temporal Annotation of French News*. Masterarbeit, Abteilung für Linguistik, Universität Georgetown 2002

[B07] Bordag, S.: *Elements of Knowledge-free and Unsupervised Lexical Acquisition*. Dissertation, Fakultät für Mathematik und Informatik, Universität Leipzig 2007

[BHL10] Butler, J.; Holden, K.; Lidwell, W.: *Universal Principles of Design*. 2. Auflage. Gloucester: Rockport Publishers 2010

[CCG+09] Calzolari, N.; Caselli, T.; Gaizauskas, R.; Hepple, M.; Im, S.; Katz, G.; Lee, K.; Nianwen, X.; Pustejovsky, J.; Saquete, E.; Saurí, R.; Schilder, F.; Verhagen, M.: *TempEval2: Evaluating Events, Time Expressions and Temporal Relations*. Proc. of the Workshop on Semantic Evaluations (SEW) 2009

[CGG+12] Cozza, R.; Glenn, D.; Gupta, A.; Lu, C.; Milanesi, C.; Nguyen, T.; Sato, A.; Shen, S.; De La Vergne, H.; Zimmermann, A.: *Market Share: Mobile Phones by Region and Country, 3Q12*. Stamford: Gartner 2012

[CPS+10] Caselli, T.; Pustejovsky, J.; Saurí, R.; Verhagen, M.: *SemEval-2010 Task 13: TempEval-2*. Proc. of the 5th International Workshop on Semantic Evaluation (SemEval) 2010

[DM06] Dale, R.; Mazur, P.: *Local Semantics in the Interpretation of Temporal Expressions*. Proc. of the Workshop on Annotating and Reasoning about Time and Events (ARTE) 2006

[DM08] Dale, R.; Mazur, P.: *What's the Date? High Accuracy Interpretation of Weekday Names*. Proc. of the 22nd International Conference on Computational Linguistics (COLING) 2008

[DM09] Dale, R.; Mazur, P.: *The DANTE Temporal Expression Tagger*. In: Human Language Technology. Challenges of the Information Society 5603. 2009

[FGM+05] Ferro, L.; Gerber, L.; Mani, I.; Sundheim, B.; Wilson, G.: *TIDES 2005 Standard for the Annotation of Temporal Expressions*. (PDF-Datei, Stand: April 2005) Internet: http://projects.ldc.upenn.edu/ace/docs/English-TIMEX2-Guidelines_v0.1.pdf (Zugriff: 15.07.2012)

[FMS+01] Ferro, L.; Mani, I.; Sundheim, B.; Wilson, G.: *Guidelines for Annotating Temporal Information*. Proc. of the Human Language Technology Conference (HLTC) 2001

[G03] Grishman, R.: *Information Extraction*. In: The Oxford Handbook of Computational Linguistics. Oxford: Oxford University Press 2003

[GKL+06] Gaizauskas, R.; Knippen, B.; Littman, J.; Pustejovsky, J.; Saurí, R.; Setzer, A.: *TimeML Annotation Guidelines*. Version 1.2.1 (PDF-Datei, Stand: 31.01.2006) Internet: http://timeml.org/site/publications/timeMLdocs/annguide_1.2.1.pdf (Zugriff: 15.07.2012)

[GS10] Gertz, M.; Strötgen, J.: *HeidelTime: High Quality Rule-based Extraction and Normalization of Temporal Expressions*. Proc. of the 5th International Workshop on Semantic Evaluation 2010

[I06] International Organization for Standardization (Herausgeber): *Data elements and interchange formats — Information interchange — Representation of dates and times*. 4. Auflage. Genf: Selbstverlag 2006

[KLP+05] Knippen, R.; Littman, J.; Pustejovsky, J.; Saurí, R.: *Temporal and Event Information in Natural Language Text*. Dordrecht: Kluwer Academic Publishers 2005

[L11] Lohnstein, H.: *Formale Semantik und natürliche Sprache*. 2., überarbeitete Auflage. Berlin: de Gruyter 2011

[LNP04] Linke, A.; Nussbaumer, M.; Portmann, P.: *Studienbuch Linguistik*. 5. Auflage. Tübingen: Niemeyer 1991

[LPS+06] Littman, J.; Pustejovsky, J.; Saurí, R.; Verhagen, M.: *TimeBank 1.2 Documentation*. 2. Version. 2006

[M74] Montague, R.: *The Proper Treatment of Quantification in Ordinary English*. In: Approaches to Natural Language. Dordrecht: Reidel 1974

[P80] Porter, M.: *An algorithm for suffix stripping*. In: Program 14(3). 1980

[R47] Reichenbach, H.: *Elements of Symbolic Logic*. Neue Auflage. New York: Dover Publications 1980

[RR10] Rappoport, A.; Reichart, R.: *Tense Sense Disambiguation: a New Syntactic Polysemy Task*. Proc. of the Conference on Empirical Methods in Natural Language Processing (EMNLP) 2010

Abbildungsverzeichnis

Tabellenverzeichnis

Selbständigkeitserklärung

Ich versichere, dass ich die vorliegende Arbeit selbständig und nur unter Verwendung der angegebenen Quellen und Hilfsmittel angefertigt habe, insbesondere sind wörtliche oder sinngemäße Zitate als solche gekennzeichnet. Mir ist bekannt, dass Zuwiderhandlung auch nachträglich zur Aberkennung des Abschlusses führen kann.

Ort Datum Unterschrift